



Swordfish NVMe Model Overview and Mapping Guide

Version: 1.2.5a

Abstract: The Swordfish NVMe Model Overview and Mapping Guide defines the model to manage NVMe and NVMeoF storage systems with Redfish and Swordfish. It provides the detailed mapping information between the NVMe, NVMeoF specifications and the Redfish and Swordfish specifications.

SNIA Approved Publication

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

Last Updated: 20 June 2023

Contents

USAGE	13
DISCLAIMER	14
Current Revision	14
Contact SNIA	14
FEEDBACK AND INTERPRETATIONS	14
INTENDED AUDIENCE	14
VERSIONING POLICY	14
Revision History	15
About SNIA	16
Acknowledgements	16
1 Abstract	18
2 Scope	19
2.1 Document Goals	19
2.2 Audience Assumptions	19
3 Normative References	20
3.1 Overview	20
3.2 Approved references	20
3.3 References under development	21
3.4 Other references	21
4 NVMe Model Overview	22
4.1 Introduction	22
4.1.1 Fundamental Model Design Assertions	22
4.2 Overall NVMe Subsystem Model	22
4.2.1 Major NVM Objects Mapped to RF/SF	23
4.2.2 Unmapped objects	23
4.2.3 NVM Subsystem Model	24
4.2.4 NVMe-oF Subsystem Model	24
5 Example Instances	26
5.1 Introduction	26
5.2 Simple SSD	26
5.2.1 Overview	26
5.2.2 Explanation of Object use	27
5.2.3 Redfish / Swordfish Object Representation	27
5.2.4 Mockup	27
5.3 Complex SSD	28
5.3.1 Overview	28

5.3.2	Explanation of Object use	28
5.3.3	Redfish / Swordfish Object Representation	29
5.4	Simple SSD with IP (NVMe-oF) Attach	29
5.4.1	Overview	29
5.4.2	Explanation of Object use	30
5.4.3	Redfish / Swordfish Object Representation	30
5.4.4	Mockup	31
5.5	JBOF	32
5.5.1	Overview	32
5.5.2	Explanation of Object use	33
5.5.3	Redfish / Swordfish Object Representation	33
5.5.4	Mockup	34
5.6	EBOF	35
5.6.1	Overview	35
5.6.2	Explanation of Object use	35
5.6.3	Redfish / Swordfish Object Representation	35
5.6.4	Mockup	36
5.7	Opaque Array / NVMe Front End Device	37
5.7.1	Overview	37
5.7.2	Explanation of Object use	37
5.7.3	Redfish / Swordfish Object Representation	38
5.7.4	Mockup	38
5.8	Subsystem (Fabric) Model - NVMe-oF: Fabric Attach Subsystem	38
5.8.1	Overview	38
5.8.2	Explanation of Object use	39
5.8.3	Redfish / Swordfish Object Representation	40
5.8.4	Mockup	40
5.9	NVMe Domains	40
5.9.1	Overview	40
5.9.2	Explanation of Object use	41
5.9.3	Mockup	41
6	Property Mapping	42
6.1	Introduction	42
6.2	Property Mapping Template	42
6.3	NVM subsystem	44
6.3.1	Mockup	44
6.3.2	Property Mapping	45
6.4	NVM Controllers	64
6.4.1	Admin Controller	64
6.4.2	Discovery Controller	98

6.4.3	IO Controller	131
6.5	Namespace	181
6.5.1	Mockup	181
6.5.2	Property Mapping	182
6.6	Endurance Group	220
6.6.1	Mockup	220
6.6.2	Property Mapping	221
6.7	NVM Set	248
6.7.1	Mockup	248
6.7.2	Property Mapping	250
6.8	Drive	276
6.8.1	Mockup	276
6.8.2	Property Mapping	278
7	Other Feature Mapping	321
7.1	Introduction	321
7.2	Firmware Update	321
7.2.1	Firmware update for NVMe Drives	321
7.3	Security Management Protocols	322
Appendix A: Objects without a direct mapping to the NVMe model		323
A.1:	Overview	323
A.2:	Related Use Cases	323
Annex B: Bibliography		325
B.1	Overview	325
B.2	Informational references	325

List of Tables

1	Revision History	15
2	Contributors	16
3	Approved normative references	20
4	Actions.#StorageController.SetEncryptionKey mapping	45
5	Controllers mapping	46
6	Description mapping	46
7	Drives mapping	48
8	Identifiers mapping	49
9	Identifiers.DurableNameFormat mapping	50
10	Identifiers.DurableName mapping	51
11	Links.Enclosures mapping	52
12	Links.Enclosures@odata.count mapping	53
13	Links.HostingStorageSystems mapping	54
14	Links.NVMeoFDiscoverySubsystems mapping	55
15	Links.SimpleStorage mapping	56
16	Name mapping	57
17	Status.State mapping	58
18	Status.Health mapping	59
19	Status.HealthRollup mapping	60
20	StorageControllers mapping	61
21	StorageGroups mapping	62
22	Volumes mapping	63
23	Actions.#StorageController.RunSelfTest mapping	66
24	Assembly mapping	67
25	Assembly mapping	68
26	CacheSummary mapping	69
27	ControllerRates mapping	70
28	Description mapping	71
29	Identifiers mapping	73
30	Identifiers.DurableName mapping	74
31	Identifiers.DurableNameFormat mapping	75
32	Links.AttachedVolumes mapping	76
33	Links.NetworkDeviceFunctions mapping	77
34	Location mapping	78
35	NVMeControllerProperties.ControllerType mapping	81
36	NNVMeControllerProperties.ANACCharacteristics mapping	89
37	SKU mapping	92
38	SpeedGbps mapping	93
39	Status.State mapping	95

40	SupportedControllerProtocols mapping	96
41	SupportedDeviceProtocols mapping	97
42	Assembly mapping	100
43	AssetTag mapping	101
44	CacheSummary mapping	102
45	ControllerRates mapping	103
46	Description mapping	104
47	FirmwareVersion mapping	105
48	Identifiers mapping	106
49	Identifiers.DurableName mapping	107
50	Identifiers.DurableNameFormat mapping	108
51	Links.AttachedVolumes mapping	109
52	Links.Endpoints mapping	110
53	Links.Connections mapping	111
54	Links.NetworkDeviceFunctions mapping	112
55	Links.NVMeDiscoveredSubsystems mapping	113
56	Location mapping	114
57	Manufacturer mapping	115
58	Model mapping	116
59	Name mapping	117
60	NVMeControllerProperties.ControllerType mapping	118
61	NVMeControllerProperties.NVMeVersion mapping	119
62	NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive SupportsTrafficBasedKeepAlive mapping	120
63	NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState SupportsExceedingPowerOfNonOperationalState mapping	121
64	NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId Supports128BitHostId mapping	122
65	NVMeControllerProperties.MaxQueueSize mapping	123
66	NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemsDegraded OverallSubsystemDegraded mapping	124
67	NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut SpareCapacityWornOut mapping	125
68	NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed mapping	126
69	Status.Health mapping	127
70	Status.State mapping	128
71	SupportedControllerProtocols mapping	129
72	SupportedDeviceProtocols mapping	130
73	Actions.#StorageController.RunSelfTest mapping	133
74	Assembly mapping	133
75	Assembly mapping	135

76	CacheSummary mapping	136
77	ControllerRates mapping	137
78	Description mapping	138
79	FirmwareVersion mapping	138
80	Identifiers mapping	139
81	Identifiers.DurableName mapping	139
82	Identifiers.DurableNameFormat mapping	140
83	Links.AttachedVolumes mapping	141
84	Links.Endpoints mapping	142
85	Links.Connections mapping	143
86	Links.NetworkDeviceFunctions mapping	143
87	Location mapping	144
88	Manufacturer mapping	144
89	Model mapping	145
90	Name mapping	146
91	NVMeControllerProperties.ControllerType mapping	147
92	NVMeControllerProperties.NVMeVersion mapping	148
93	NVMeControllerProperties.NVMeControllerAttributes.ReportsUUIDList mapping	149
94	SupportsSQAssociations mapping	150
95	NVMeControllerProperties.NVMeControllerAttributes.	151
96	NVMeControllerProperties.NVMeControllerAttributes.	152
97	SupportsTrafficBasedKeepAlive mapping	153
98	NVMeControllerProperties.NVMeControllerAttributes.	154
99	NVMeControllerProperties.NVMeControllerAttributes.	155
100	NVMeControllerProperties.NVMeControllerAttributes.SupportsReadRecoveryLevels	156
101	NVMeControllerProperties.NVMeControllerAttributes.SupportsNVMSets	157
102	SupportsExceedingPowerOfNonOperationalState	158
103	Supports128BitHostId	159
104	NVMeControllerProperties.MaxQueueSize mapping	160
105	NNVMeControllerProperties.ANCharacteristics mapping	161
106	NNVMeControllerProperties.ANCharacteristics.AccessState mapping	162
107	NNVMeControllerProperties.ANCharacteristics.Volume mapping	163
108	NNVMeControllerProperties.NVMeSMARTCriticalWarnings.PRMunreliable mapping	164
109	NVMeControllerProperties.NVMeSMARTCriticalWarnings.	165
110	NVMeControllerProperties.NVMeSMARTCriticalWarnings.MediaInReadOnly mapping	166
111	OverallSystemDegraded mapping	167
112	NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut	168
113	PCIeInterface.PCIeType mapping	169
114	PCIeInterface.MaxPCIeType mapping	170
115	PCIeInterface.LanesInUse mapping	171
116	PCIeInterface.MaxLanes mapping	172

117	Ports mapping	173
118	SKU mapping	174
119	SpeedGbps mapping	175
120	Status.State mapping	176
121	Status.Health mapping	177
122	SupportedControllerProtocols mapping	179
123	SupportedDeviceProtocols mapping	180
124	BlockSizeBytes mapping	183
125	Capacity.Data.ConsumedBytes mapping	185
126	Capacity.Data.ProvisionedBytes mapping	186
127	Capacity.Data.AllocatedBytes mapping	188
128	Capacity.Metadata.AllocatedBytes mapping	189
129	Description mapping	190
130	DisplayName mapping	190
131	Identifiers mapping	191
132	Identifiers.DurableName mapping	192
133	Identifiers.DurableNameFormat mapping	193
134	InitializeMethod mapping	194
135	Links.Controllers	195
136	Links.Drives mapping	195
137	LogicalUnitNumber mapping	196
138	MaxBlockSizeBytes mapping	197
139	Name mapping	198
140	NVMeNamespaceProperties.NamespaceId mapping	199
141	NVMeNamespaceProperties.IsBootCapable mapping	200
142	NVMeNamespaceProperties.IsShareable mapping	201
143	NVMeNamespaceProperties.NamespaceFeatures.	202
144	NVMeNamespaceProperties.NamespaceFeatures.	203
145	NVMeNamespaceProperties.NamespaceFeatures.SupportsNGUIDReuse mapping	204
146	NVMeNamespaceProperties.NamespaceFeatures.	205
147	NVMeNamespaceProperties.NamespaceFeatures.	207
148	163: NVMeNamespaceProperties.NumberLBAFormats mapping	208
149	NVMeNamespaceProperties.FormattedLBASize mapping	209
150	NVMeNamespaceProperties.MetadataTransferredAtEndOfDataLBA mapping	210
151	NVMeNamespaceProperties.NVMeVersion mapping	211
152	OptimumIOSizeBytes mapping	212
153	OptimumIOSizeBytes mapping	214
154	Status.State mapping	215
155	Status.Health mapping	216
156	Status.HealthRollup mapping	217
157	StorageGroups mapping	218

158	WriteCachePolicy mapping	219
159	AllocatedPools mapping	222
160	Capacity.Data.AllocatedBytes mapping	223
161	Capacity.Data.ConsumedBytes mapping	224
162	CapacitySources mapping	225
163	CapacitySources@odata.count mapping	226
164	Description mapping	227
165	Links.OwningStorageResource mapping	228
166	Name mapping	229
167	NVMeProperties.NVMePoolType	230
168	NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent mapping	231
169	NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed mapping	232
170	NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate mapping	233
171	NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead mapping	235
172	NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten mapping	236
173	NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten mapping	237
174	NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount mapping	238
175	NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount mapping	239
176	NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount mapping	240
177	NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount mapping	241
178	NVMeSetProperties.SetIdentifier	242
179	NVMeSetProperties.OptimalWriteSizeBytes mapping	243
180	NVMeSetProperties.EnduranceGroupIdentifier mapping	244
181	NVMeSetProperties.Random4kReadTypicalNanoSeconds mapping	245
182	Status.Health mapping	246
183	Status.State mapping	247
184	AllocatedVolumes mapping	250
185	Capacity.Data.AllocatedBytes mapping	251
186	Capacity.Data.ConsumedBytes mapping	252
187	CapacitySources mapping	253
188	CapacitySources@odata.count mapping	254
189	Description mapping	255
190	Links.OwningStorageResource mapping	256
191	Name mapping	257
192	NVMeProperties.NVMePoolType	258
193	NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent mapping	259
194	NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed mapping	260
195	NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate mapping	261
196	NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead mapping	262
197	NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten mapping	263
198	NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten mapping	264

199	NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount mapping	265
200	NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount mapping	266
201	NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount mapping	267
202	NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount	268
203	NVMeSetProperties.SetIdentifier	268
204	NVMeSetProperties.OptimalWriteSizeBytes mapping	270
205	NVMeSetProperties.EnduranceGroupIdentifier mapping	271
206	NVMeSetProperties.Random4kReadTypicalNanoSeconds mapping	272
207	NVMeSetProperties.UnallocatedNVMNamespaceCapacityBytes mapping	273
208	Status.State mapping	273
209	Status.Health mapping	275
210	Actions.#Drive.Reset mapping	278
211	Actions.#Drive.SecureErase mapping	280
212	Assembly.BinaryDataURI mapping	281
213	BlockSizeBytes mapping	282
214	CapableSpeedGpbs mapping	284
215	CapacityBytes for single namespace mapping	285
216	CapacityBytes for multiple namespace mapping	286
217	Description mapping	287
218	EncryptionAbility mapping	288
219	EncryptionStatus mapping	289
220	FailurePredicted mapping	290
221	Identifiers mapping	291
222	Identifiers.DurableNameFormat mapping	292
223	Identifiers.DurableName mapping	293
224	IndicatorLED mapping	294
225	Links.Volume mapping	295
226	Links.Volumes@odata.count mapping	296
227	Location mapping	297
228	LocationIndicatorActive mapping	298
229	Manufacturer mapping	299
230	MediaType mapping	300
231	Metrics mapping	301
232	Model mapping	302
233	Multipath mapping	303
234	Name mapping	305
235	NegotiatedSpeedGbps mapping	306
236	PhysicalLocation.Info mapping	307
237	PhysicalLocation.InfoFormat mapping	308
238	PhysicalLocation.PartLocation mapping	309
239	PredictedMediaLifetimeLeftPercent mapping	310

240	Protocol mapping	311
241	Revision mapping	312
242	RotationSpeedRPM mapping	313
243	SKU mapping	314
244	SerialNumber mapping	315
245	Status.State mapping	316
246	Status.Health mapping	317
247	StatusIndicator mapping	319
248	WriteCacheEnabled mapping	320
249	Additional parameters	322

List of Figures

1	Subsystem model	24
2	NVMe-oF Subsystem Model	25
3	Simple SSD instance diagram	26
4	Simple SSD mockup example	27
5	Complex SSD Model	28
6	Complex SSD Model	29
7	Simple IP-attached SSD	30
8	Simple IP-attached SSD mockup	31
9	JBOF configuration controller object	32
10	Full JBOF system	33
11	JBOF system instance	34
12	Full EBOF system	35
13	EBOF system instance	36
14	Opaque array example	37
15	Sample opque system instance	38
16	NVMe-OF subsystem example	39
17	NVMe-oF system instance	40
18	NVMeDomain example	41

USAGE

Copyright (c) 2020 - 2023 SNIA. All rights reserved. All other trademarks or registered trademarks are the property of their respective owners.

The SNIA hereby grants permission for individuals to use this document for personal use only, and for corporations and other business entities to use this document for internal use only (including internal copying, distribution, and display) provided that:

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

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 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 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.

Revision History

Revisions to this document are summarized in Table 1.

Table 1: Revision History

Date	Rev	Notes
18 August 2020	1.2.1	Initial Release
31 October 2020	1.2.1c	Released as SNIA Approved Publication
2 March 2021	1.2.2	Added detailed mapping information to match information in NVMe drive profiles for many new referenced properties, and included guidance for mandatory/recommended implementation as reflected in the profiles. Added sections for firmware update, with details for NVMe Drive implementation requirements. Added cross-references to User’s Guide NVMe-specific use cases. Errata fixes – correct diagram, correct table headers.
30 August 2021	1.2.3	Update Mapping Guide with new mapping guidance, corresponding to the Swordfish NVMe Front End profile.
5 December 2021	1.2.3	Release as SNIA Approved Publication
12 April 2022	1.2.4	Release as SNIA Working Draft. Errata fixes throughout. Added section for EBOF configuration, and additions to appendix A to correspond to EBOF, JBOF, and NVMe-oF configurations.
12 July 2022	1.2.4a	Release as SNIA Standard.

Member	Representatives (* – prior employer)
	Jim Pendergraft
	Michael Raineri
Hewlett Packard Enterprise	Curtis Ballard
	Jeff Hilland
	Chris Lionetti
Intel Corporation	Richelle Ahlvers
	Rajalaxmi Angadi
	Phil Cayton
	Slawek Putyrski
Kioxia	Mark Carlson
Lenovo	Keith Campbell
NetApp, Inc.	Don Deel
	Fred Knight
Samsung Corporation	Lu Fan
	Bill Martin
	Tom Rainey
VMware, Inc.	Murali Rajagopal

1 Abstract

The Swordfish NVMe Model Overview and Mapping Guide defines the model to manage NVMe and NVMe-oF storage systems with Redfish and Swordfish. It provides the detailed mapping information between the NVMe, NVMe-oF specifications and the Redfish and Swordfish specifications.

2 Scope

2.1 Document Goals

This document describes how both the NVMe Subsystem model and the NVMe-oF fabric system model should be mapped consistently to Redfish and Swordfish constructs for implementations to be managed within Redfish and Swordfish management environments.

This model and mapping information does not describe or assert any specific implementation recommendation technologies.

This document also provides the mapping information for properties recommended to be implemented in Redfish/Swordfish for NVMe and NVMe-oF devices and the corresponding reference information from the NVMe and NVMe-oF specifications.

2.2 Audience Assumptions

This document assumes that the reader of this document is familiar with NVMe and NVMe-oF technologies and concepts. It also assumes the reader has knowledge of the Redfish and Swordfish concepts.

3 Normative References

3.1 Overview

The documents listed in Table 3 are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

3.2 Approved references

Table 3: Approved normative references

Tag	Title (Version)	Author	URL
ISO-8601	Data elements and interchange formats – Information interchange – Representation of dates and times – Part 1: Basic rules	ISO / IEC	< http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=70907 >
ISO -Direct	ISO/IEC Directives, Part 2 Principles and rules for the structure and drafting of ISO and IEC documents (Eighth Edition, 2018)	ISO / IEC	< https://www.iso.org/sites/directives/current/part2/index.xhtml >
Redfish	Redfish Scalable Platforms Management API Specification (v1.11.0)	DMTF	http://www.dmtf.org/sites/default/files/standards/documents/DSP0266_1.4.0.pdf
Swordfish	Swordfish Scalable Storage Management API Specification (v1.2.5)	SNIA	< http://www.snia.org/tech_activities/standards/curr_standards/swordfish >
NVMe-Base	NVM Express Base Specification 2.0c	NVM Express	https://nvmexpress.org/developers/nvme-specification/
NVMe-Command	NVM Express NVM Command Set Specification 1.0c	NVM Express	https://nvmexpress.org/developers/nvme-command-set-specifications/

Tag	Title (Version)	Author	URL
NVMe-Transport	NVMe over PCIe Transport Specification 1.0c	NVM Express	https://nvmexpress.org/developers/nvme-transport-specifications/
NVMe-MI	NVMe-MI 1.2c Specification	NVM Express	https://nvmexpress.org/developers/nvme-mi-specification/
SPC-5	SCSI Primary Commands - 5	INCITS	https://webstore.ansi.org/standards/incits/incits5022019

3.3 References under development

None defined in this document.

3.4 Other references

None defined in this document.

4 NVMe Model Overview

4.1 Introduction

In order to manage NVMe and NVMe-oF devices and systems in a large scale environment, a higher level management ecosystem is needed. The Redfish/Swordfish management specifications are designed to manage multi-system environments, including multiple types of fabrics, covering not only multiple technologies, but also inclusive of system management, storage management and fabric management, making it the ideal ecosystem in which to add not only the integration of NVMe devices for system and storage management, but NVMe-oF for fabric management.

This document describes how both the NVMe Subsystem model and the NVMe-oF fabric system model should be mapped consistently to Redfish and Swordfish constructs for implementations to be managed within Redfish and Swordfish management environments. This model and mapping information does not describe or assert any specific implementation recommendation technologies.

Similar implementations will have similar Redfish and Swordfish constructs. Mockups are used to show static examples of sample representations. Requirements and recommendations for implementations are provided separately through the Swordfish NVMe and NVMe-oF profiles. The profiles use the Redfish interoperability profile schema to specify the required, recommended and optional properties and schema for specific configurations and functionality that correspond to classes of implementations.

4.1.1 Fundamental Model Design Assertions

- There shall be a unified model across all types of NVMe devices.
- There shall not be a different model for “drives” vs other types of NVMe devices
- The model will cover an appropriate level of abstraction for all types of NVMe devices based on modeling and mock-ups reflected in the documented permutations (e.g., from simple drives through to complex fabric virtual systems)
 - Simple NVMe drives; complex NVMe drives; JBOFs/EBOFs; Arrays/RBOFs
- The logical model for NVMe-oF shall leverage the NVMe Subsystem model
- Logical subsystems, controllers, and namespaces are the same objects with the same relationships as in the NVMe Subsystem Model. Unneeded objects are not instantiated (e.g., Endurance Groups, sets)
- The NVMe native model should map to the existing Redfish and Swordfish constructs when and where possible
 - Mapping future NVMe / NVMe-oF functionality should follow this principle when and where possible (e.g., firmware update mapping to the RF update service)

4.2 Overall NVMe Subsystem Model

Key Tenets:

- Model reflects a unified view of all NVMe device types.
- Devices will instantiate an appropriate subset of the model
- The model diagrams do not reflect all available schema elements.
- Model leverages and coarsely maps to existing (Redfish and) Swordfish storage model

4.2.1 Major NVM Objects Mapped to RF/SF

4.2.1.1 NVM Subsystem An NVM subsystem includes one or more controllers, zero or more namespaces, and one or more ports. Examples of NVM subsystems include Enterprise and Client systems that utilize PCI Express based solid state drives and/or fabric connectivity.

4.2.1.2 NVM Controller (IO, Admin and Discovery) The interface between a host and an NVM subsystem

Admin controller: controller that exposes capabilities that allow a host to manage an NVM subsystem

Discovery: controller that exposes capabilities that allow a host to retrieve a Discovery Log Page

I/O: controller that implements I/O queues and is intended to be used to access a non-volatile memory storage medium

4.2.1.3 Namespace 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 $(n-1)$.

4.2.1.4 Endurance Group A portion of NVM in the NVM subsystem whose endurance is managed as a group

4.2.1.5 NVM Set An NVM Set is a collection of NVM that is separate (logically and potentially physically) from NVM in other NVM Sets.

4.2.1.6 NVM Domain A domain is the smallest indivisible unit that shares state (e.g., power state, capacity information). Domain members can be NVM controllers, endurance groups, sets or namespaces.

4.2.2 Unmapped objects

There are a number of objects that are required for the proper integration of NVMe support within Redfish and Swordfish, but which are not supported by an entity that can be mapped directly from the various NVM Specifications. Information about these related but un-mapped objects are defined elsewhere in the documentation provided with each Swordfish release.

They are summarized in Appendix A.

4.2.3 NVM Subsystem Model

The following diagram reflects the high level mapping of the key NVM objects into Redfish / Swordfish schema objects. These largely follow existing relationships used by the Swordfish storage specification for non-NVMe implementations as well, which provides a great deal of consistency for storage clients, as well as for implementations such as NVMe arrays that may be delivering solutions that combine NVMe and other technologies.

This model covers a wide range of instantiations ranging from individual SSDs, to multi-rack storage systems. All of these can be represented by this NVM Subsystem model, shown in Figure 1.

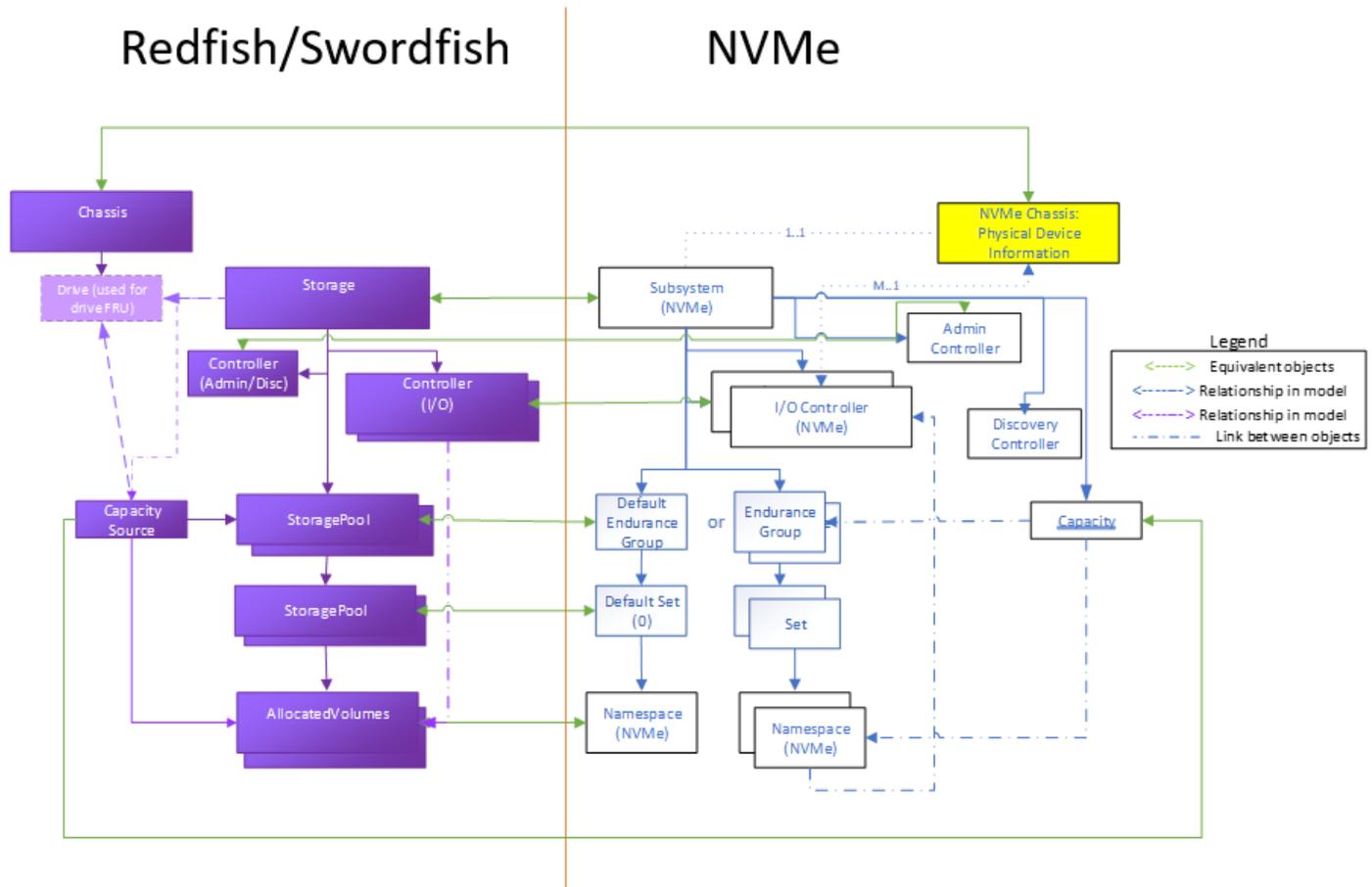


Figure 1: Subsystem model

4.2.4 NVMe-oF Subsystem Model

Figure 2 shows the high level mapping of the key NVMe-oF objects to Redfish / Swordfish schema objects. Following the tenets described in the model overview section, these extend the mapping used in the NVM Subsystem for the logical versions of the objects.

This model also includes the use of the Redfish Fabric model to cover the connectivity aspects of the fabric.

The grey shaded portion of this diagram reflects the logical / exported portion of the NVMe-oF environment represented in Redfish / Swordfish.

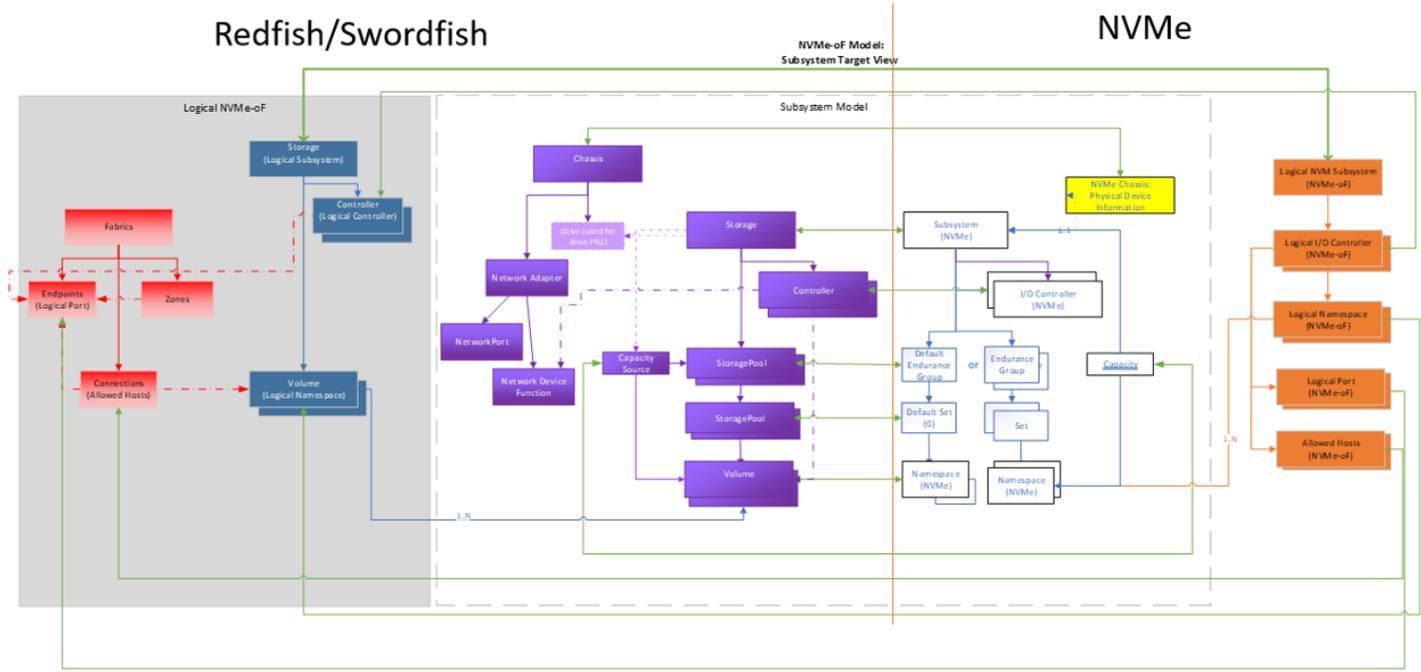


Figure 2: NVMe-oF Subsystem Model

5 Example Instances

5.1 Introduction

This section of the document provides a series of example usages of the model that represent common instantiations of NVMe devices, and how those devices may use the various NVMe, and correspondingly, Redfish and Swordfish objects and schema. This section will not provide comprehensive representations of all potential device types; rather, a representation of several common device types, in order to provide an illustration of the application of the model for those that are unfamiliar with either the NVMe or Redfish / Swordfish ecosystems.

Further, the following sections describe the examples and do not attempt to cover all potential permutations for alternate representations of each device class or possible implementations.

5.2 Simple SSD

5.2.1 Overview

Figure 3 shows a sample representation of a simple NVMe SSD, with a PCIe interface. It is implemented with no endurance group or NVM set functionality; it has only a single namespace capability, and a single IO controller.

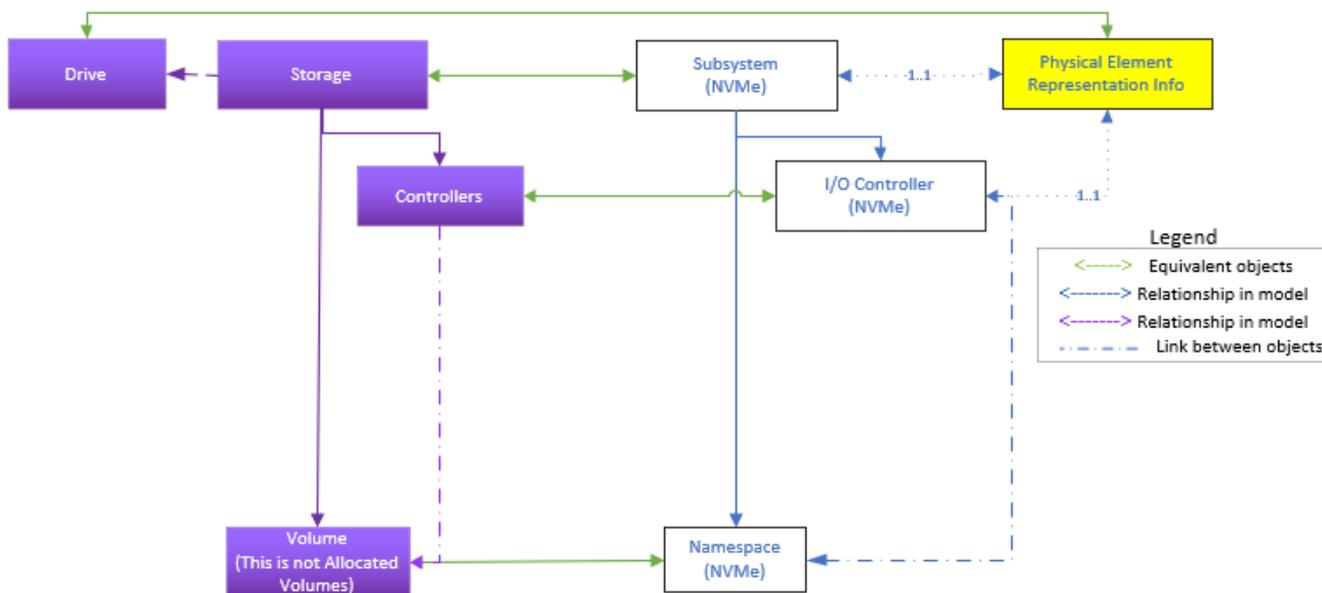


Figure 3: Simple SSD instance diagram

5.2.2 Explanation of Object use

Simple SSDs are SSDs that do not use Endurance Groups or sets. Correspondingly, they only use Storage, Controllers, Volumes (Namespaces), and the Drive schema to represent the fundamental components.

Many SSD implementations support exactly one namespace. These are described by this model, and the mockups reflect this configuration.

This model can also support extensions to cover dual-ported configurations, as well as support for multiple IO controllers per port.

5.2.3 Redfish / Swordfish Object Representation

Figure 4 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

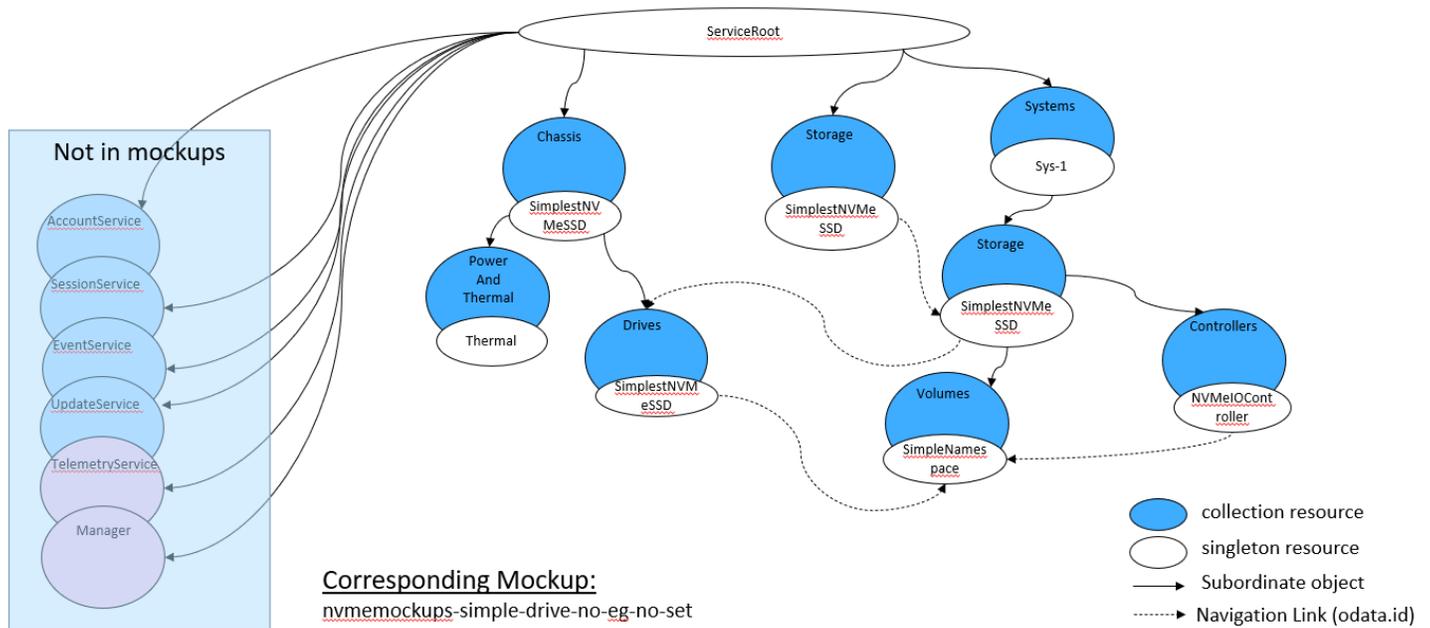


Figure 4: Simple SSD mockup example

5.2.4 Mockup

A corresponding mockup for this configuration can be found at <http://simple-ssd-mockups.swordfishmockups.com/redfish/v1/>.

5.3.3 Redfish / Swordfish Object Representation

Figure 6 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

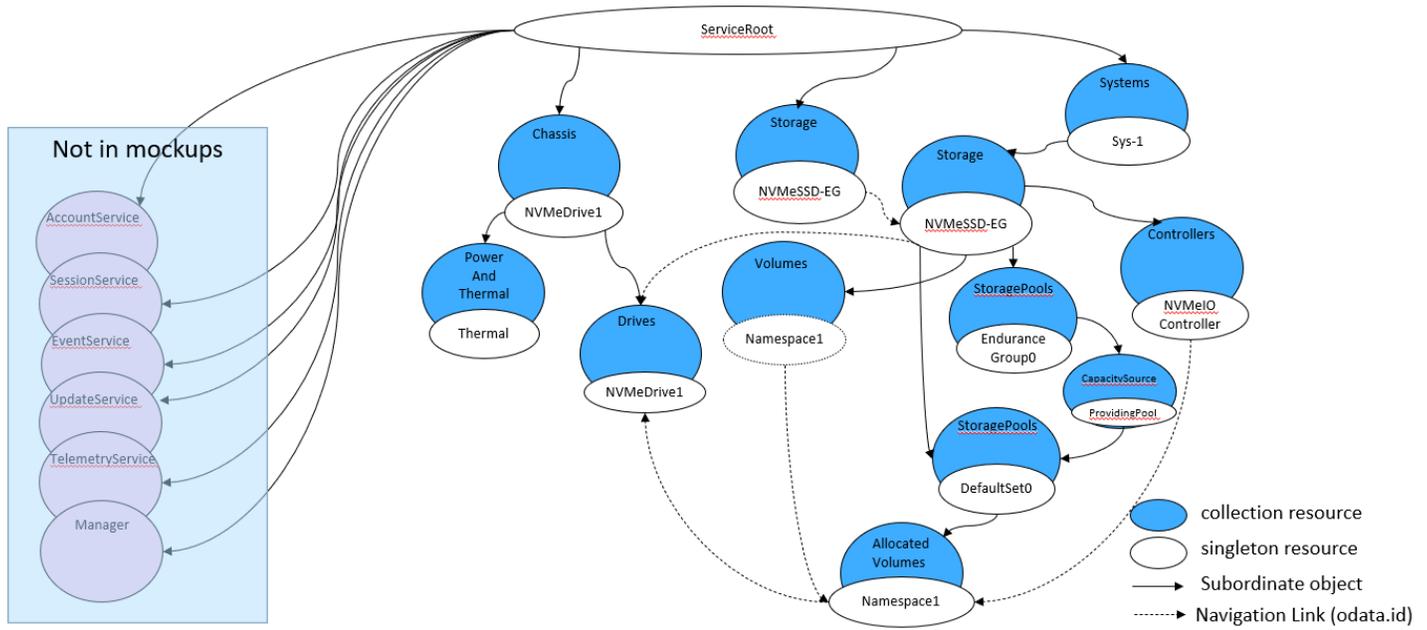


Figure 6: Complex SSD Model

5.3.3.1 Mockup A corresponding mockup for this configuration can be found at <http://simple-ssd-eg-set-mockups.swordfishmockups.com/redfish/v1/>.

5.4 Simple SSD with IP (NVMe-oF) Attach

5.4.1 Overview

This example reflects an IP-attached drive configuration, with a single ethernet port configured, as illustrated in Figure 7. It includes a drive configured with a default endurance group and NVM Set, and is instantiated in the Storage Collection off the Service Root. The network configuration is modeled in the Chassis.

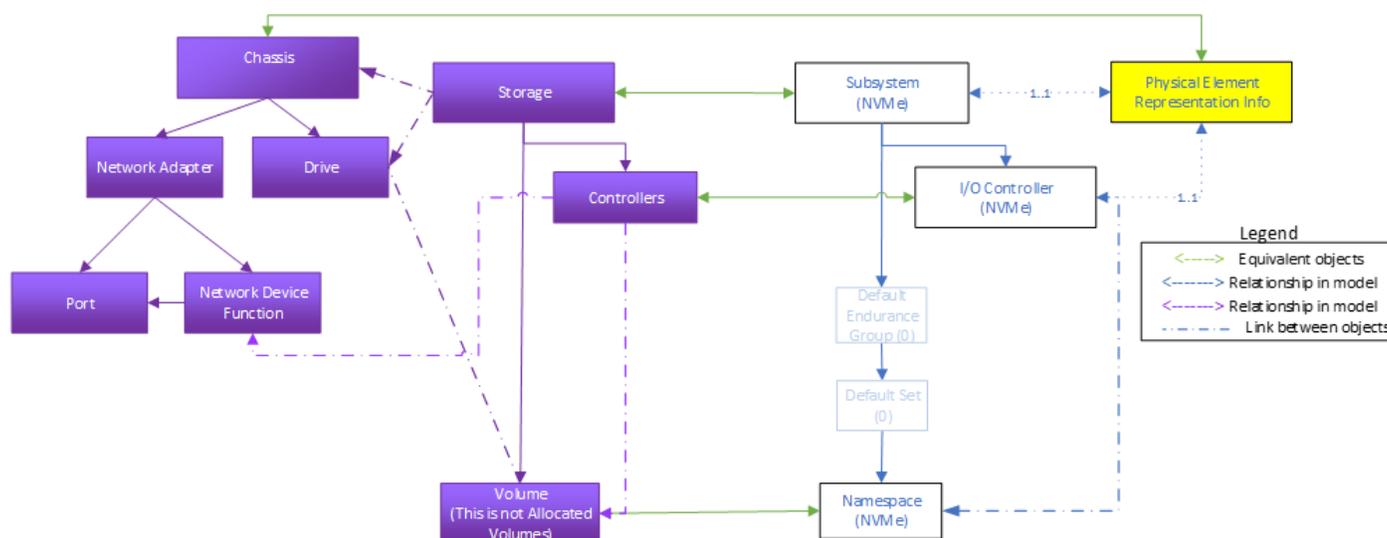


Figure 7: Simple IP-attached SSD

5.4.2 Explanation of Object use

Simple SSDs with IP attach are also SSDs that do not use Endurance Groups or sets, but that have IP-based network interfaces. Correspondingly, they only use Storage, Controllers, Volumes (Namespaces), and the Drive schema to represent the fundamental components. In addition, they use the Redfish Network Adapter, Port and Network Device Function to model the configuration of the IP interface port(s).

As with the Simple SSD configuration, Many SSD with IP-attach implementations support exactly one namespace. These are described by this model, and the mockups reflect this configuration.

This model can also support extensions to cover multi-ported configurations, as well as support for multiple IO controllers per port.

5.4.3 Redfish / Swordfish Object Representation

Figure 8 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

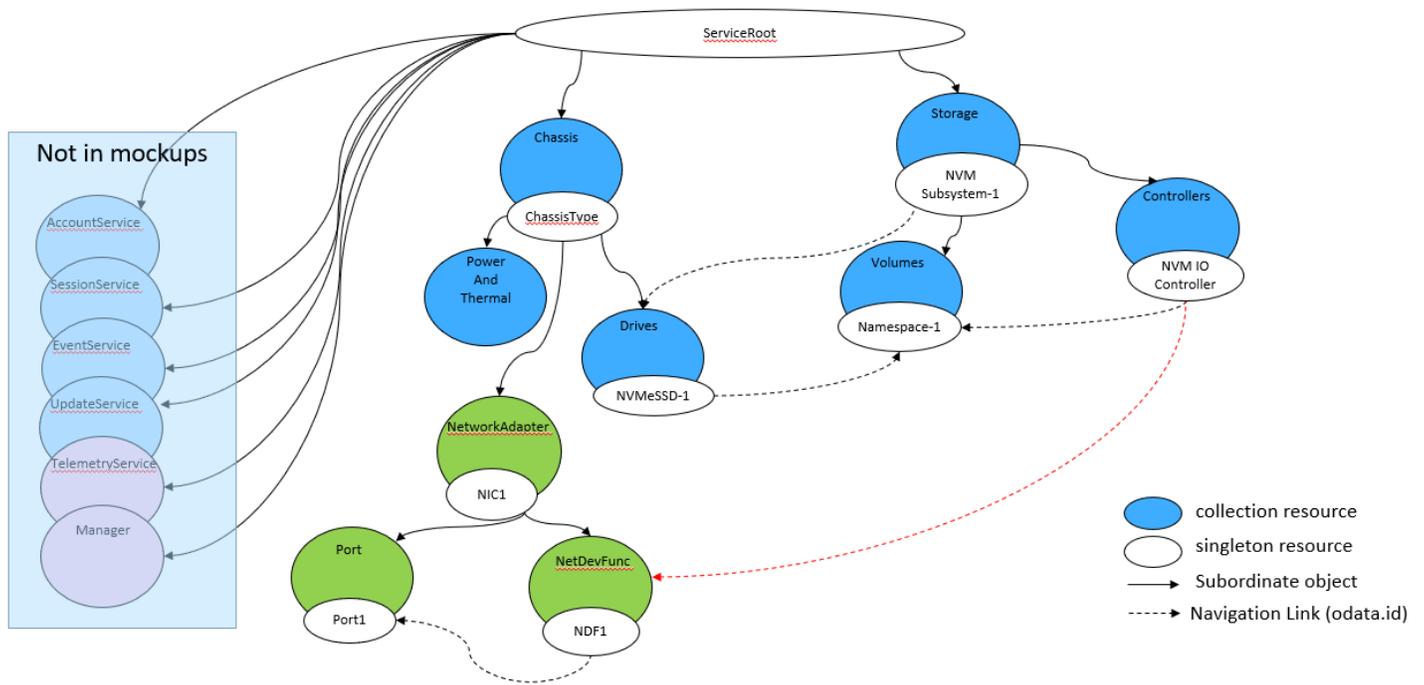


Figure 8: Simple IP-attached SSD mockup

5.4.4 Mockup

A corresponding mockup for this configuration can be found at <http://ethernet-attach-drive-mockups.swordfishmockups.com/redfish/v1/>.

5.5 JBOF

5.5.1 Overview

This example covers a representation of a JBOF (“just a bunch of flash”) enclosure and contained drives. This mockup reflects a PCIe front-end attach configuration with a set of drives.

Figure 9 shows only the controller object representation for this JBOF configuration. This includes the admin controller function for enclosure management.

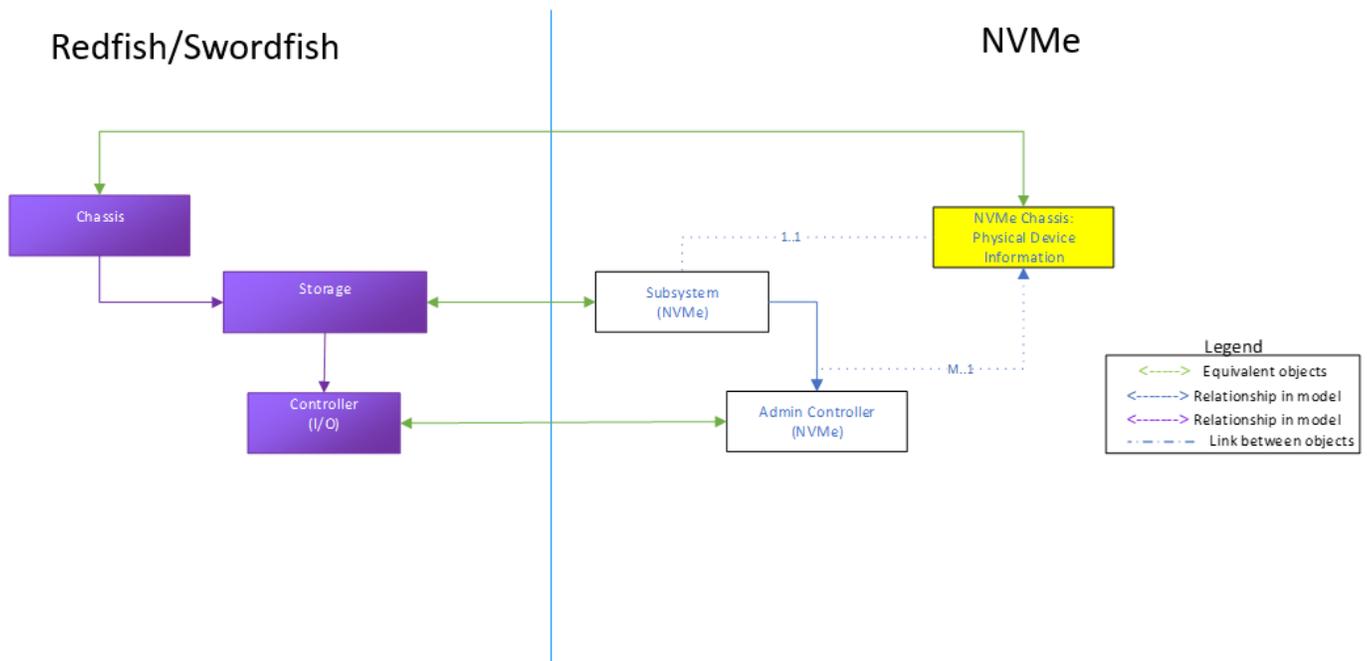


Figure 9: JBOF configuration controller object

Figure 10 shows the combined object representations for the JBOF system, with both the JBOF controller and NVMe drives (using the Simple SSD style drives) represented in the system. Note that the full mockup represented has 7 drives, while this diagram only represents two for the sake of visual clarity.

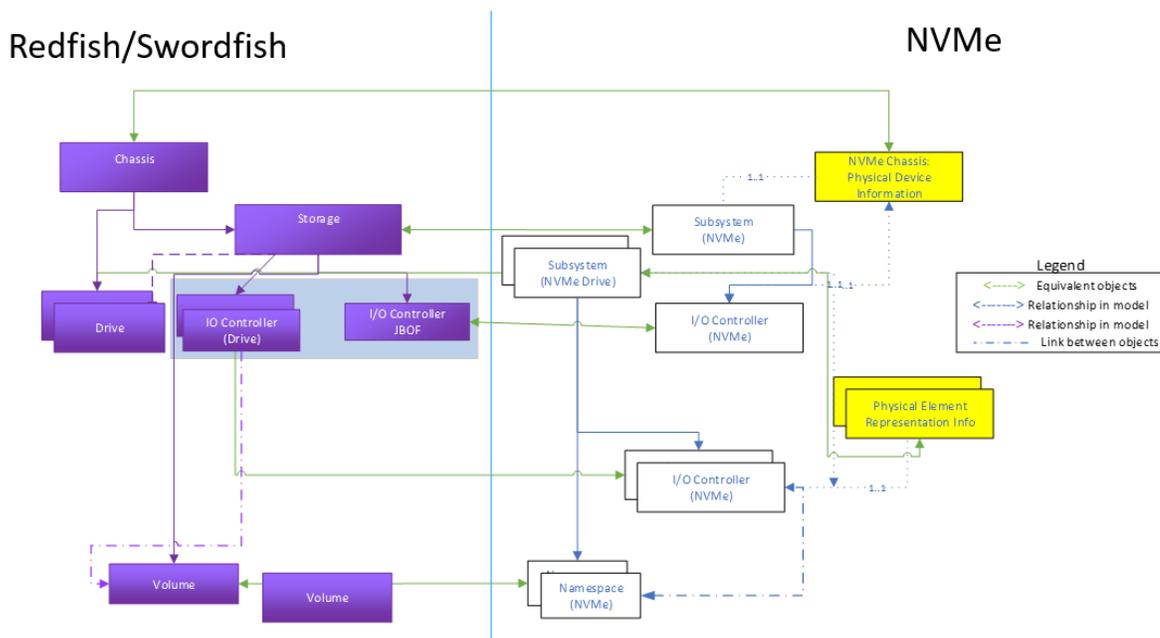


Figure 10: Full JBOF system

5.5.2 Explanation of Object use

This type of JBOF system uses the Chassis, Storage and Controller objects to reflect physical component modeling, Sub-system and Admin Controller functionality.

The Chassis model and Admin controller represent NVM’s SES (SCSI enclosure services) usage.

5.5.3 Redfish / Swordfish Object Representation

Figure 11 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

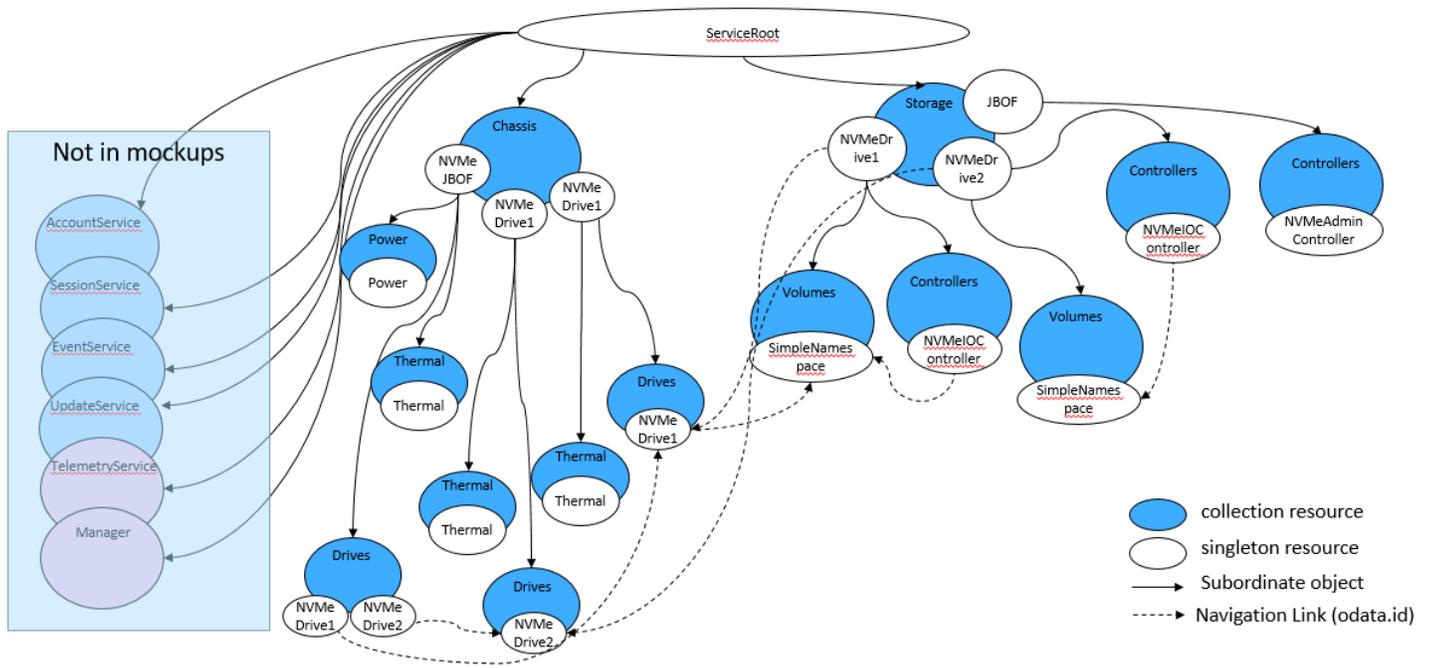


Figure 11: JBOF system instance

5.5.4 Mockup

A corresponding mockup for this configuration can be found at <http://nvme-jbof-mockups.swordfishmockups.com/redfish/v1/>

5.6 EBOF

5.6.1 Overview

This example covers a representation of a EBOF (“Ethernet bunch of flash”) enclosure and contained drives. This mockup reflects a Ethernet front-end attach enclosure configuration containing a set of Ethernet-attach drives.

Figure 12 shows the combined object representations for the EBOF system, with both the EBOF controller and Ethernet-attach drives (using the Simple SSD with IP drives) represented in the system. Note that the full mockup represented has multiple drives, while this diagram only represents one for the sake of visual clarity.

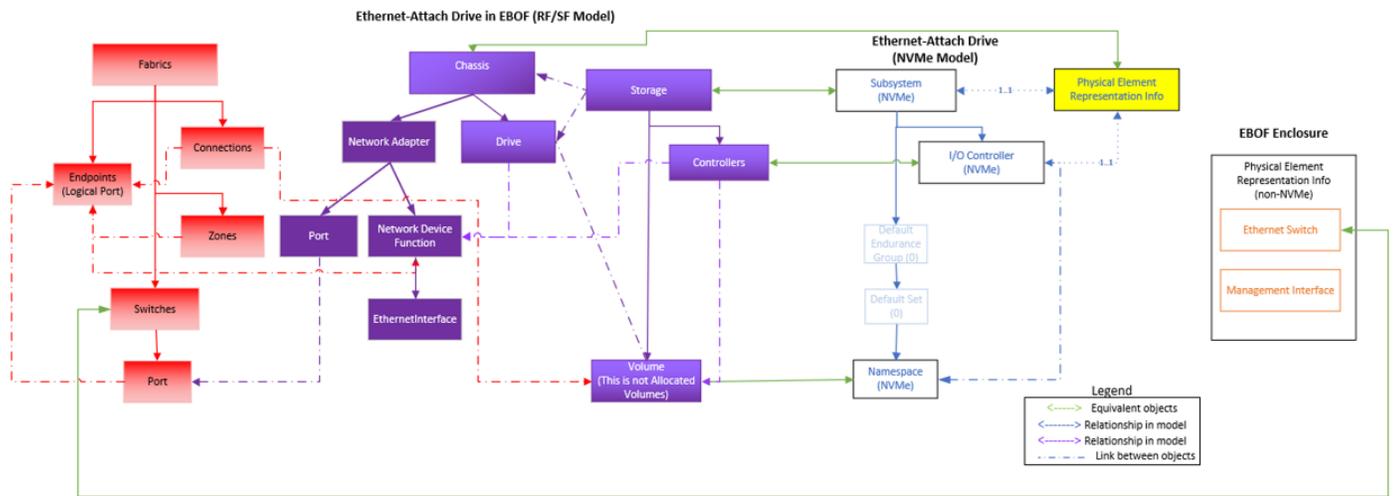


Figure 12: Full EBOF system

5.6.2 Explanation of Object use

This type of EBOF system uses the Storage and Controller objects to reflect component modeling of the NVMe Subsystem and controller functionality.

Connectivity is represented through the Chassis model’s NetworkAdapter, NetworkPort, NetworkDeviceFunction and EthernetInterface objects, the Fabric model’s Switch and Port objects, as well as the Manager model’s EthernetInterface and ManagerProtocol. In-band management capability is also modeled via Admin controllers, consistent with other NVMe devices.

5.6.3 Redfish / Swordfish Object Representation

Figure 13 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

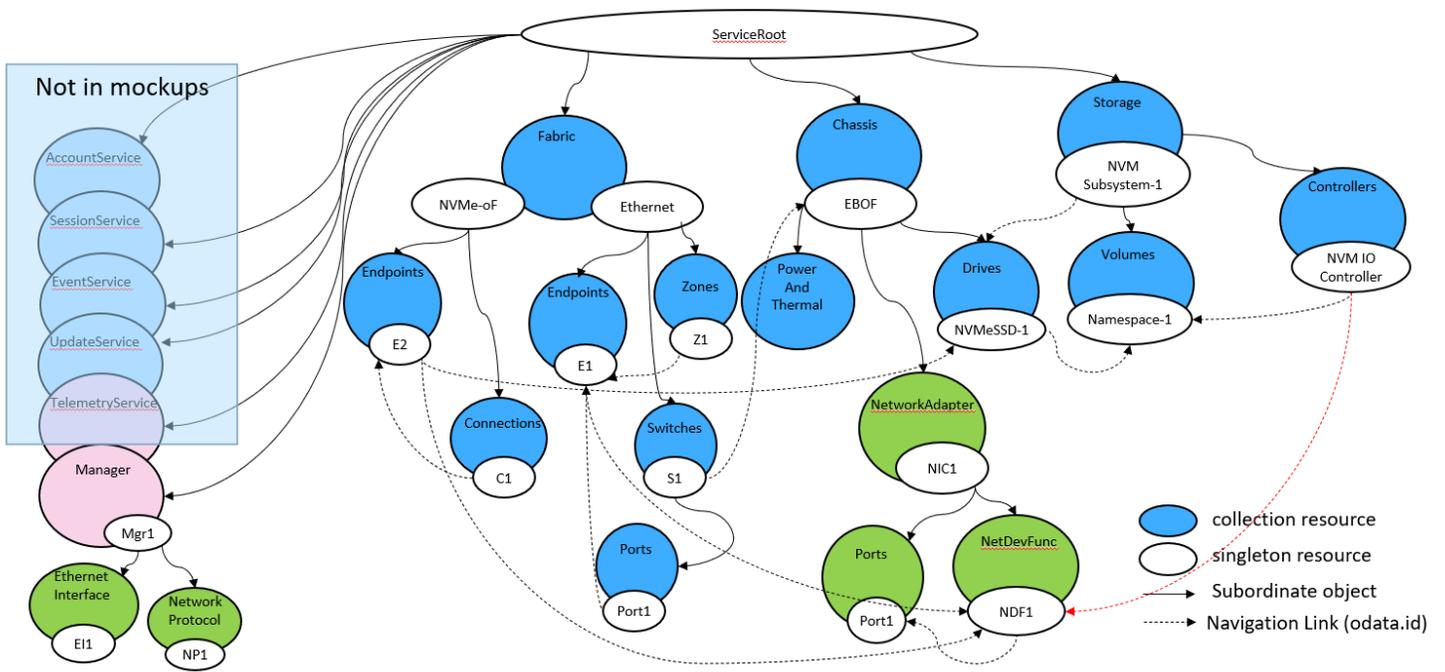


Figure 13: EBOF system instance

5.6.4 Mockup

A corresponding mockup for this configuration can be found at <http://nvme-ebof-mockups.swordfishmockups.com/redfish/v1/>

5.7.3 Redfish / Swordfish Object Representation

Figure 15 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

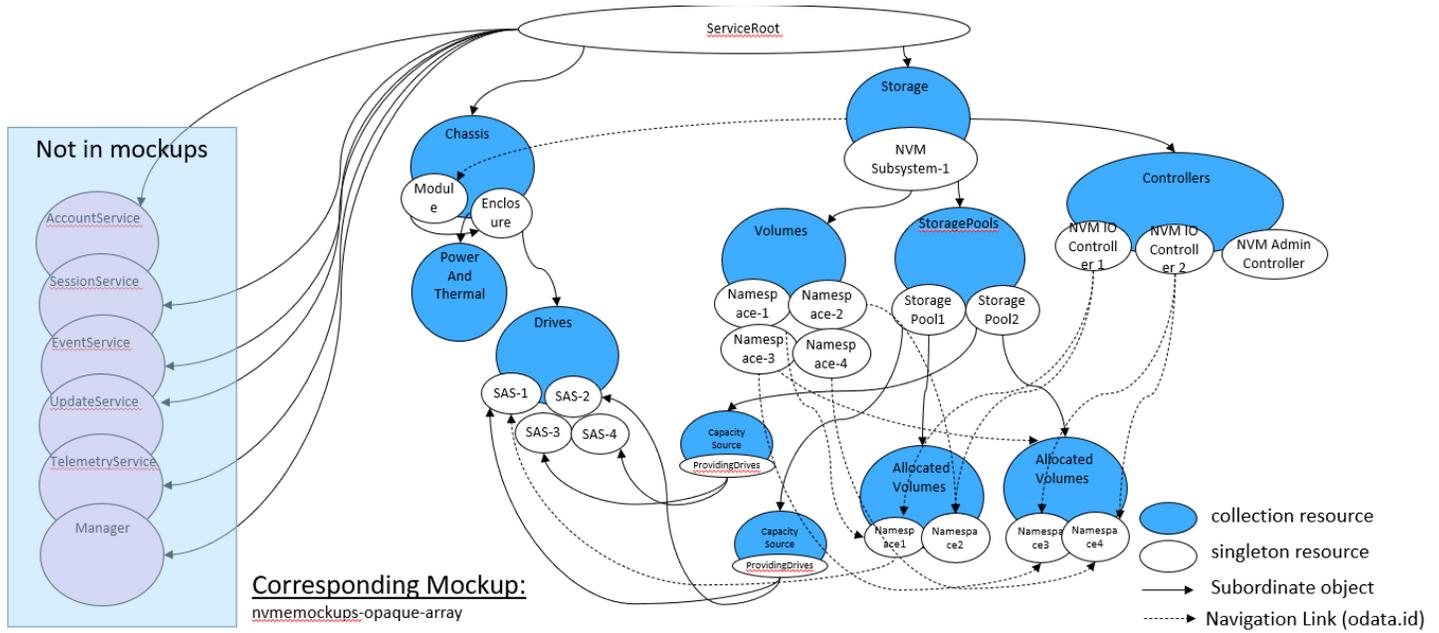


Figure 15: Sample opaque system instance

5.7.4 Mockup

A corresponding mockup for this configuration can be found at <http://nvme-opaque-array-mockups.swordfishmockups.com/redfish/v1/>.

5.8 Subsystem (Fabric) Model - NVMe-oF: Fabric Attach Subsystem

5.8.1 Overview

Figure 16 shows a logical/exported NVMe-oF subsystem presenting one logical subsystem, one I/O controller, one namespace, one port and representing one allowed host, using the Redfish Fabric model. The fabric model uses the Connection schema to characterize the allowed host information, and Endpoints and Zones to show the network connectivity from the device’s perspective.

This example also includes NVMeDomains. NVMeDomains contain a collection of domain members; these can be NVM controllers, endurance groups, NVM sets, or namespaces.

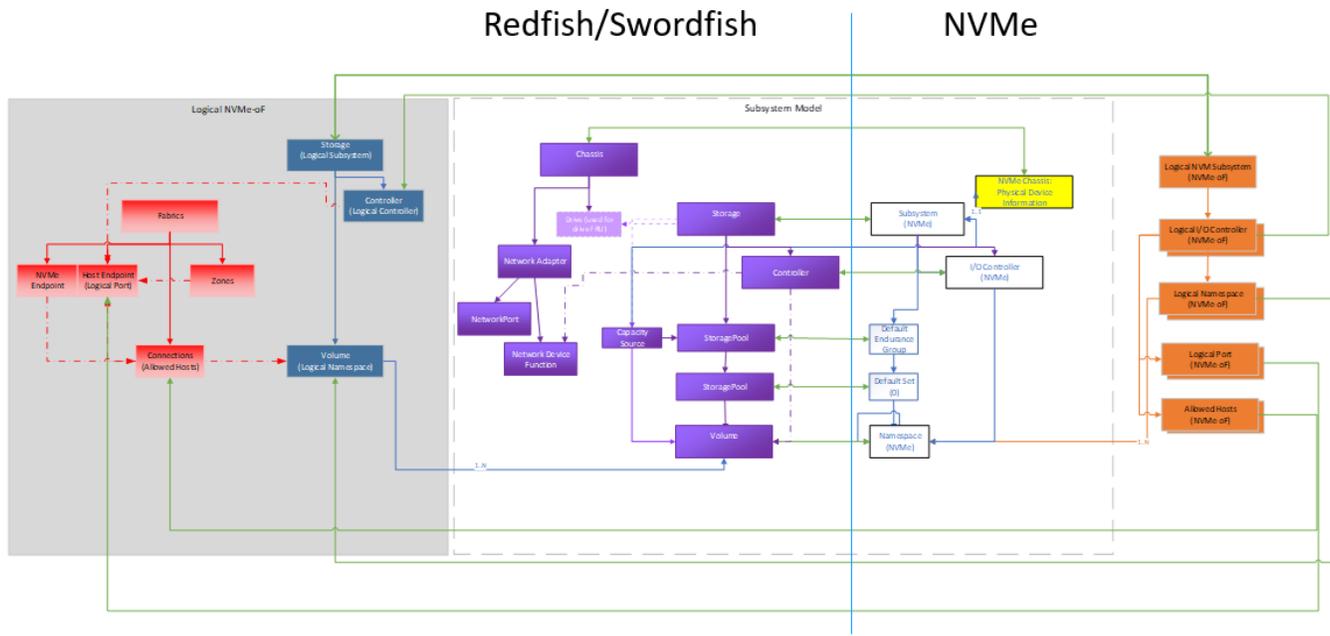


Figure 16: NVMe-OF subsystem example

5.8.2 Explanation of Object use

The fabric attach subsystem example shows the representation of logical, or exported, subsystems, controllers, and namespaces using the Storage, Controller and Volume objects respectively.

It also uses portions of the Redfish fabric model to represent the host attachment information - allowed hosts and logical port - using the Connections and Endpoints objects respectively.

5.8.3 Redfish / Swordfish Object Representation

Figure 17 shows the representation, as expressed in the mockup indicated below, of a sample instantiation using Redfish / Swordfish objects.

Note that this mockup does not represent a complete service instantiation; it contains only objects of interest for this context.

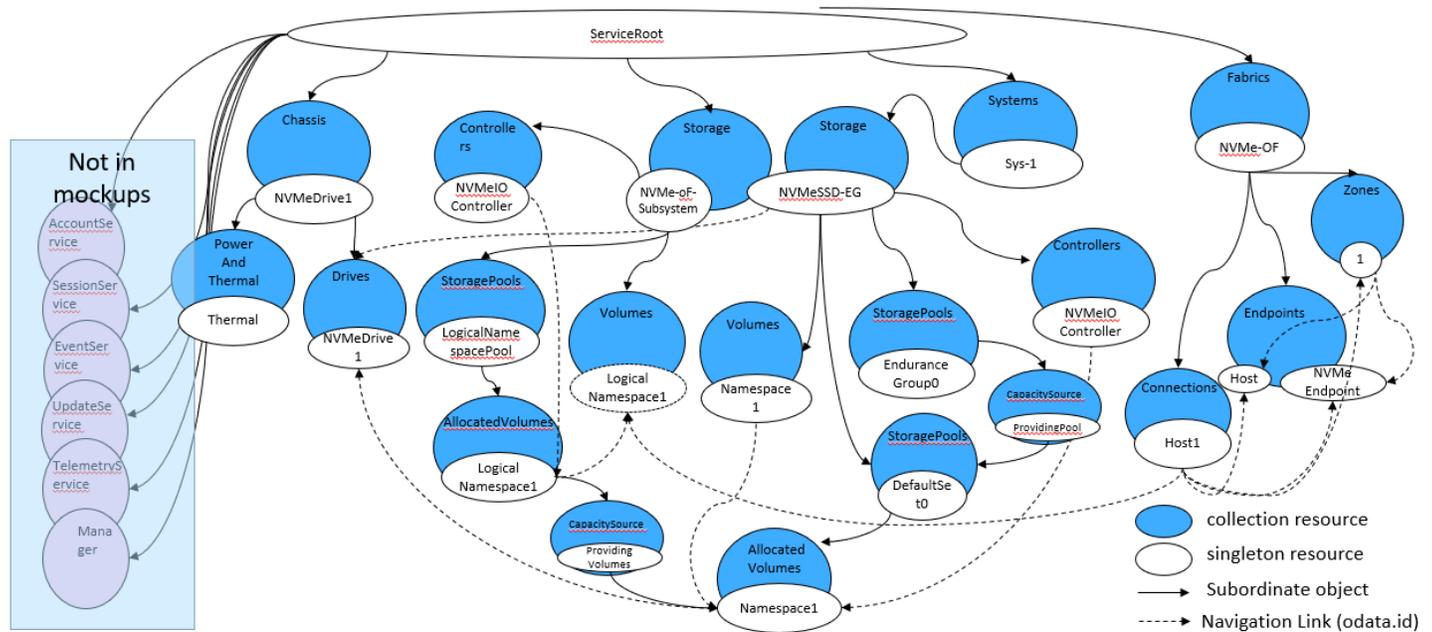


Figure 17: NVMe-oF system instance

5.8.4 Mockup

There are multiple mockups that show the representation for this configuration:

- A mockup for this configuration using an Ethernet attach front-end can be found at <http://nvmeof-mockups.swordfishmockups.com/redfish/v1/>.
- A mockup for this configuration using an RDMA attach front-end can be found at <http://nvmeof-RDMA-mockup.swordfishmockups.com/redfish/v1/>.

5.9 NVMe Domains

5.9.1 Overview

This example describes NVMeDomains. NVMeDomains contain a collection of domain members; these can be NVM controllers, endurance groups, NVM sets, namespaces, and ports, as illustrated in Figure 18.

Domains are used to subdivide an NVM Subsystem. For example, if there are multiple power sources, the domain is used to represent the scope of each power source.

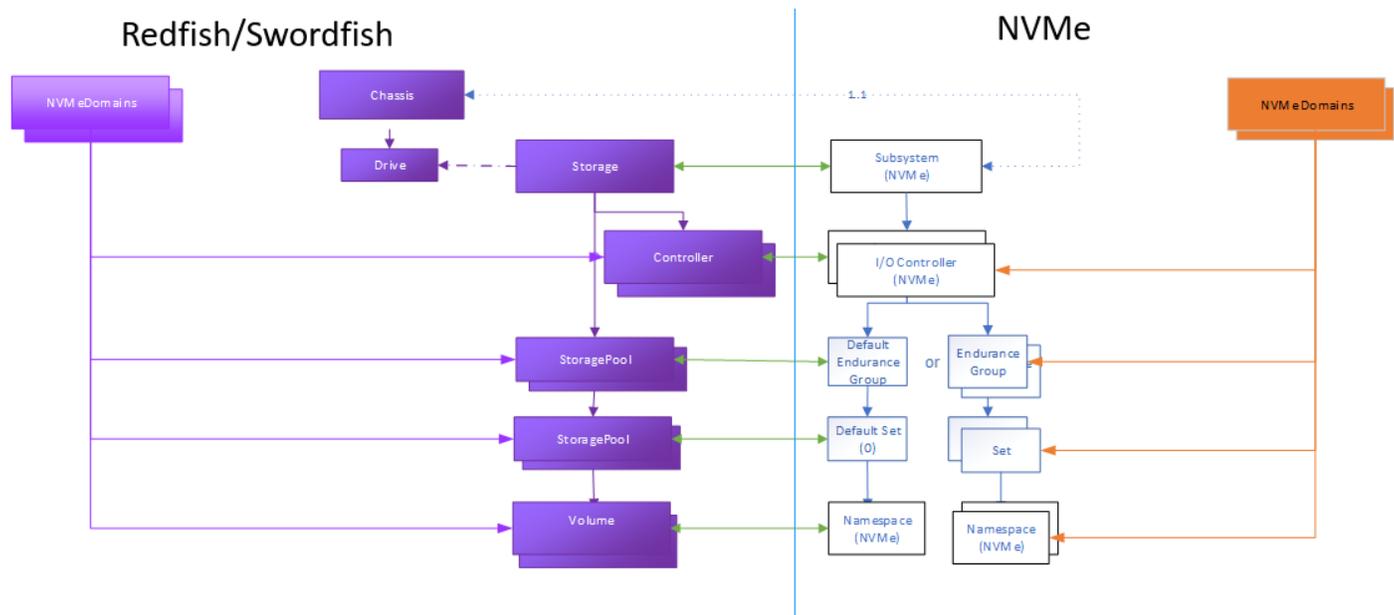


Figure 18: NVMeDomain example

5.9.2 Explanation of Object use

The domain object contains two primary elements: the domain members collection, which contains pointers to the relevant controllers, groups, sets, namespaces, and ports that reflect the appropriate subdivision for the purpose of the domain.

The other primary element includes a set of capacity information properties about this set of domain members.

5.9.3 Mockup

A corresponding mockup for this configuration can be found at <http://nvmeof-mockups.swordfishmockups.com/redfish/v1/>.

6 Property Mapping

6.1 Introduction

The property mapping provided defines the preferred translation between the Redfish/Swordfish schema objects and properties, and the corresponding NVMe and NVMe-oF specification properties. The information is ordered within the Redfish/Swordfish objects; each cross-referenced property within the Redfish/Swordfish structure therefore contains a detailed reference to its mapped property in the NVMe ecosystem.

Each section also includes a sample Redfish/Swordfish mockup presenting an example usage for that object.

6.2 Property Mapping Template

Table 6.2 provides the template and an example for the property mapping provided in the following sections of this document.

For each property (whether a reference, collection, complex type or actual property), there is a comparison between the property in Redfish/Swordfish to the corresponding property in either the NVMe or NVMe-oF specification. The RF/SF property is provided within its schema context; the NVMe/NVMe-oF specification reference is provided within the table, showing both which specification, as well as the section and, if appropriate, the figure in which the property is specified.

Similarly, the type of each property is correspondingly specified. The RF/SF type is specified, and the NVM Spec property type is shown, as well as, where appropriate, any additional identifying information, such as byte offset and data structure.

The Mandatory field is used to specify whether properties are Mandatory or Optional on the NVMe specification side, and in the rare instance where properties are Mandatory in the Redfish/Swordfish schema. (Recommended / required properties for specific implementation types in Redfish/Swordfish will be done separately, through the use of profiles.)

The Notes field can / will be used to include any relevant information about either the purpose of the property, additional context, or other useful information to implementers, such as inter-relationships with other properties.

```
+--+-----+-----+ | | Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
+--+-----+-----+
```

: Property Mapping Template and Example

Property | **Redfish / Swordfish Schema Property:**

RecommendedArbitrationBurstSize| **NVM Spec Property / Field:** Identify Command / Identify Controller (CNS 01h) / RecommendedArbitrationBurst(RAB)

NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1, Figure 275 (Identify Controller Data Structure, I/O Command Set Independent) / RecommendedArbitrationBurst(RAB): Bytes 72 || Type | **Redfish / Swordfish Schema Type:** String | Power of 2^n | | Description | The Recommended Arbitration Burst Size indicates the maximum number of commands that the controller may launch at one time from a particular Submission Queue. | This is the recommended Arbitration Burst size. || LongDescription | This property shall contain the Recommended Arbitration Burst Size indicates the maximum number of commands that the controller may launch at one time from a particular Submission Queue. The value is expressed as a power of two (e.g., 000b indicates one, 011b indicates eight). A value of 111b indicates no limit. | This is the recommended Arbitration Burst size. The value is in commands and is reported as a power of two (2^n). This is the same units as the Arbitration Burst size. || Mandatory | Mandatory | I/O Controller: Mandatory

Admin Controller: Mandatory

Discovery Controller: Restricted || Notes ||

6.3 NVM subsystem

The Redfish/Swordfish Storage schema is used to represent an NVM Subsystem.

6.3.1 Mockup

The following mockup shows a sample representation of the Storage schema used to represent an NVM Subsystem.

```
{
  "@odata.id": "/redfish/v1/Storage/NVMeoF-SS1",
  "@odata.type": "#Storage.v1_15_0.Storage",
  "Id": "1",
  "Name": "NVMe-oF Logical NVM Fabric System",
  "Description": "An NVM Express Subsystem is an NVMe device that contains one or more NVM Express controllers and may contain one or more namespaces.",
  "Status": {
    "State": "Enabled",
    "Health": "OK",
    "HealthRollup": "OK"
  },
  "Identifiers": [{
    "DurableNameFormat": "NQN",
    "DurableName": "nqn.2014-08.org.nvmexpress:uuid:6c5fe566-10e6-4fb6-aad4-8b4159f50245"
  }],
  "Controllers": {
    "@odata.id": "/redfish/v1/Storage/NVMeoF-SS1/Controllers"
  },
  "Volumes": {
    "@odata.id": "/redfish/v1/Storage/NVMeoF-SS1/Volumes"
  },
  "Links": {
    "NVMeoFDiscoverySubsystems": [{
      "@odata.id": "/redfish/v1/Storage/NVMeoF-Discovery"
    }]
  },
  "@odata.id": "/redfish/v1/Storage/NVMeoF-SS1",
  "@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."
}
```

6.3.2 Property Mapping

6.3.2.1 Actions.#StorageController.SetEncryptionKey The mapping for `Actions.#StorageController.SetEncryptionKey` is summarized in Table 4.

Table 4: `Actions.#StorageController.SetEncryptionKey` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Actions.#StorageController.SetEncryptionKey</code>	N/A
Type	Action (Special form of POST)	N/A
Description	The available OEM-specific actions for this resource.	N/A
LongDescription	This property shall contain the available OEM-specific actions for this resource.	N/A
Mandatory	Optional Do Not Implement for NVMe Drives, or for devices with an NVMe front end interface, e.g., opaque arrays.	DNI
Notes		N/A for NVMe (drives or for devices with an NVMe front end interface). Drives will generate their own key for CryptoErase, this requires passing a new key.

6.3.2.2 Controllers The mapping for `Controllers` is summarized in Table 5.

Table 5: Controllers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Controllers	NVM Spec Property / Field: Identify Command / Controller List (CNS 13h) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.12
Type	StorageControllerCollection.	Controller list.
Description	The set of controllers instantiated by this storage subsystem.	A List of controllers that exist in the NVM subsystem.
LongDescription	This property shall contain a link to a Resource of type StorageControllerCollection that contains the set of storage controllers allocated to this instance of an storage subsystem.	A Controller List of up to 2,047 controller identifiers is returned containing a controller identifier greater than or equal to the value specified in the Controller Identifier (CDW10.CNTID) field. The list contains controller identifiers in the NVM subsystem that may or may not be attached to namespace(s). Reference Section 4.4.1, Figure 134 (Controller List Format)
Mandatory	Mandatory.	No (see note)
Notes	This is a collection of StorageControllers. Refer to the StorageController schema for details of the instance information. These are used to provide information on NVM IO, Admin and Discovery controllers.	This property is only mandatory for controllers that support the Namespace Management capability.

6.3.2.3 Description The mapping for `Description` is summarized in Table 6.

Table 6: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Description is a read-only field.	Return the common description: "An NVM Express Subsystem is an NVMe device that contains one or more NVM Express controllers and may contain one or more namespaces."

6.3.2.4 Drives

The mapping for `Drives` is summarized in Table 7.

Table 7: Drives mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Drives	N/A
Type	Collection(Drive.Drive)	N/A
Description	The set of drives attached to the storage controllers that this resource represents.	N/A
LongDescription	This property shall contain a set of the drives attached to the storage controllers that this resource represents.	N/A
Mandatory	Optional Mandatory for NVMe Drives.	
Notes	For NVMe Drive implementation, this links to “Drive” object, which contains the physical representation of NVMe Drive information.	

6.3.2.5 Identifiers The mapping for `Identifiers` is summarized in Table 8.

Table 8: Identifiers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers	N/A
Type	Collection(Resource.Identifier)	N/A
Description	The Durable names for the subsystem.	N/A
LongDescription	This property shall contain a list of all known durable names for the associated subsystem.	N/A
Mandatory	Mandatory	
Notes	This is an array of unique identifiers for the NVM Subsystem. There will only be one instance in this array for Subsystem. Refer to the Identify Controller data structure (CNS 01h) bits 1023:768 in figure 275 (Identify – Identify Controller Data Structure, I/O Command Set Independent) of the NVMe Base Specification.	

6.3.2.6 Identifiers.DurableNameFormat

The mapping for `Identifiers.DurableNameFormat` is summarized in Table 9.

Table 9: `Identifiers.DurableNameFormat` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Identifiers.DurableNameFormat</code>	NVM Spec Property / Field: NVMe Qualified Names NVM Spec: Section:Figure NVMe 2.0: Section 4.5
Type	<code>Resource.v1_1_0.DurableNameFormat</code>	UTF-8 null-terminated string
Description	The format of the Durable names for the subsystem.	This specifies the NVM Subsystem NVMe Qualified Name as a UTF-8 null-terminated string.
LongDescription	This specifies the format of the associated NVM Subsystem NVMe Qualified Name of type NQN. Support for this field is mandatory if the controller supports revision 1.2.1 or later as indicated in the Version register (refer to section 3.1.2).	This specifies the NVM Subsystem NVMe Qualified Name as a UTF-8 null-terminated string.
Mandatory	Optional	Yes if the controller supports revision 1.2.1 or later as indicated in the Version register.
Notes	This is an enum with multiple potential values. For this particular usage in Subsystem, there will only be one instance populated, of type NQN which is a UTF-8 null-terminated string.	

6.3.2.7 Identifiers.DurableName

The mapping for `Identifiers.DurableName` is summarized in Table 10.

Table 10: Identifiers.DurableName mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableName	NVM Spec Property / Field: Identify Command / Identify Controller (CNS 01h) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1, Figure 275 (Identify Controller Data Structure, I/O Command Set Independent), Bits 768-1023 - NVM Subsystem NVMe Qualified Name (SUBNQN)
Type	Edm.String	UTF-8 null-terminated string
Description	The format of the Durable names for the subsystem.	This field specifies the NVM Subsystem NVMe Qualified Name (SUBNQN)
LongDescription	This specifies the NVM Subsystem NVMe Qualified Name as a UTF-8 null-terminated string.	This field specifies the NVM Subsystem NVMe Qualified Name as a UTF-8 null-terminated string. Refer to NVMe Base Specification, section 4.5, for the definition of NVMe Qualified Name.
Mandatory	Optional	Yes if the controller supports revision 1.2.1 or later as indicated in the Version register (refer to section 3.1.3.2).
Notes	For this particular usage in Subsystem, there will only be one instance populated in the identifiers array.	

6.3.2.8 Links.Enclosures

The mapping for `Links.Enclosures` is summarized in Table 11.

Table 11: Links.Enclosures mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.Enclosures	N/A
Type	Collection(Chassis.Chassis)	N/A
Description	An array of links to the chassis to which this storage subsystem is attached.	N/A
LongDescription	This property shall contain an array of links to resources of type Chassis that represent the physical containers attached to this resource.	N/A
Mandatory	Mandatory	
Notes	For NVMe Drive implementation, this links to a chassis collection that contains the subsystem's "Drive" object, which contains the physical representation of NVMe Drive information. For devices with an NVMe front end interface, e.g., (opaque arrays), this refers to the appropriate chassis instance for the device/system; there may be multiple chassis instances, reflecting different physical entities in the system, such as controllers, drive enclosures, racks, etc).	The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.3.2.9 Links.Enclosures@odata.count The mapping for `Links.Enclosures@odata.count` is summarized in Table 12.

Table 12: `Links.Enclosures@odata.count` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.Enclosures@odata.count</code>	N/A
Type	<code>(odata property)int64</code>	N/A
Description	Count of the number of items in the <code>Links.Enclosures</code> array.	N/A
LongDescription		N/A
Mandatory	Mandatory	
Notes		The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.3.2.10 Links.HostingStorageSystems

The mapping for `Links.HostingStorageSystems` is summarized in Table 14.

Table 13: Links.HostingStorageSystems mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.HostingStorageSystems	N/A
Type	Collection(ComputerSystem.ComputerSystem)	N/A
Description	The storage systems that host this storage subsystem.	N/A
LongDescription	This property shall contain an array of links to resources of type ComputerSystem that represent the storage systems that host this storage subsystem. The members of this array shall be in the StorageSystems resource collection off the service root.	N/A
Mandatory	Optional Recommended for devices with an NVMe front end interface such as opaque arrays.	
Notes	For devices with an NVMe front end interface, e.g., opaque arrays), this refers to the ComputerSystem instances providing the NVMe front end, modeling the device's controller(s).	The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.3.2.11 Links.NVMeoFDiscoverySubsystems The mapping for `Links.NVMeoFDiscoverySubsystems` is summarized in Table 15.

Table 14: `Links.NVMeoFDiscoverySubsystems` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links .NVMeoFDiscoverySubsystems</code>	N/A
Type	<code>Collection(Storage.Storage)</code>	N/A
Description	An array of links to the discovery subsystems that discovered this subsystem in an NVMe-oF environment.	N/A
LongDescription	This property shall contain an array of links to resources of type <code>Storage</code> that represent the discovery subsystems that discovered this subsystem in an NVMe-oF environment.	N/A
Mandatory	Optional Recommended for devices with an NVMe front end interface such as opaque arrays; required for use in NVMe-oF environments and environments using discovery controllers.	
Notes	This information should be constructed from the Swordfish service. There is not a known mapping to NVMe-oF information at this time.	

6.3.2.12 Links.SimpleStorage The mapping for `Links.SimpleStorage` is summarized in Table 15.

Table 15: `Links.SimpleStorage` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.SimpleStorage</code>	N/A
Type	<code>SimpleStorage.SimpleStorage</code>	N/A
Description	The link to the simple storage instance that corresponds to this storage.	N/A
LongDescription	This property shall contain a link to a resource of type <code>SimpleStorage</code> that represents the same storage subsystem as this resource.	N/A
Mandatory	Do Not Implement	
Notes		

6.3.2.13 Name The mapping for `Name` is summarized in Table 16.

Table 16: Name mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Name	N/A
Type	String	N/A
Description	The name of the resource or array member.	N/A
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Name is a read-only field.	

6.3.2.14 Status.State

The mapping for `Status.State` is summarized in Table 17.

Table 17: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	N/A
Type	Resource.State (enum)	N/A
Description	The known state of the resource, such as, enabled.	N/A
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	N/A
Mandatory	Optional Recommended to implement for NVMe front end devices such as opaque arrays.	
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySpare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified	In general, there is no simple corresponding property or mappable set of information at this time from the NVMe Specifications. Current guidance for NVMe drives: do not implement this property. Guidance will be added in a future version of this document as this is an important concept for clients and for consistency with traditional storage devices. For opaque arrays and other similar devices with an NVMe front end, support and map these properties to the device's concepts of Enabled/Disabled/etc.

6.3.2.15 Status.Health

The mapping for `Status.Health` is summarized in Table 18.

Table 18: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	NVM Spec Property / Field: Smart / Health Information Log (Log Identifier 02h) / Critical Warning / the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3, Figure 207 (Get Log Page – SMART / Health Information Log), Byte 00 (Critical Warning), Bit 02
Type	Resource.Health	8 Bit value
Description	The health state of this resource in the absence of its dependent resources.	Indicates if the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	Indicates if the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability
Mandatory	Mandatory	Yes
Notes	Possible Values: OK / Warning / Critical	Implementations of more complex systems, such as opaque arrays and other similar devices with an NVMe front end, may also map this property to the device's concepts of OK/Warning/Critical.

6.3.2.16 Status.HealthRollup The mapping for `Status.HealthRollup` is summarized in Table 19.

Table 19: `Status.HealthRollup` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Status.HealthRollup</code>	N/A
Type	<code>Resource.Health</code>	N/A
Description	The overall health state from the view of this resource.	N/A
LongDescription	This property shall represent the health state of the resource and its dependent resources. The values shall conform to those defined in the Redfish Specification.	N/A
Mandatory	Mandatory	

Notes | Possible Values: OK / Warning / Critical | Reference the section in the Swordfish Specification regarding `Status.HealthRollup`

6.3.2.17 StorageControllers The mapping for `StorageControllers` is summarized in Table 20.

Table 20: `StorageControllers` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>StorageControllers</code>	N/A
Type	<code>Storage.StorageControllers</code>	N/A
Description	The set of storage controllers that this resource represents.	N/A
LongDescription	This property shall contain a set of the storage controllers that this resource represents.	N/A
Mandatory	Do Not Implement	
Notes	Deprecated for NVMe use - replaced by <code>Controllers</code> (type <code>StorageController.StorageController</code>).	

6.3.2.18 StorageGroups

The mapping for `StorageGroups` is summarized in Table 21.

Table 21: StorageGroups mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	StorageGroups	N/A
Type	StorageGroup.StorageGroup	N/A
Description	All of the storage groups, each of which contains a set of volumes and endpoints that are managed as a group for mapping and masking, that belong to this storage subsystem.	N/A
LongDescription	This property shall contain a link to a resource collection of type <code>StorageGroupsCollection</code> . This property shall be used when implementing mapping and masking.	N/A
Mandatory	Do Not Implement	
Notes	N/A for NVMe use cases. Deprecated by Connections.	

6.3.2.19 Volumes The mapping for `VoLumes` is summarized in Table 22.

Table 22: Volumes mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Volumes	N/A
Type	VolumeCollection.VolumeCollection	N/A
Description	The set of volumes instantiated by this storage subsystem.	N/A
LongDescription	This property shall contain a link to a Resource of type VolumeCollection that contains the set of storage volumes allocated to this instance of an storage subsystem.	N/A
Mandatory	Optional Required for NVMe drives, as well as opaque arrays and other similar devices with an NVMe front end.	
Notes	This is a collection of Namespaces that belong to this NVM Subsystem. Refer to the Volume schema for details of the instance information.	

6.4 NVM Controllers

The Redfish/Swordfish StorageControllers schema is used to represent an NVM Controller.

There are three different types of NVM Controllers: Admin, Discovery, and IO.

6.4.1 Admin Controller

6.4.1.1 Mockup The following mockup shows a sample representation of the StorageController schema used to represent an Admin Controller.

```
{
  "@odata.type": "#StorageController.v1_3_0.StorageController",
  "Name": "NVMe Admin Controller",
  "Description": "Single NVMe Admin Controller for in-band admin command access.",
  "Status": {
    "State": "Enabled",
    "Health": "OK"
  },
  "Id": "NVMeAdminController",
  "Model": "NVMe Connect Array",
  "SerialNumber": "NVME123456",
  "PartNumber": "NVM44",
  "FirmwareVersion": "1.0.0",
  "SupportedControllerProtocols": [
    "PCIe"
  ],
  "NVMeControllerProperties": {
    "ControllerType": "Admin",
    "NVMeVersion": "1.3",
    "NVMeControllerAttributes": {
      "SupportsSQAssociations": false,
      "SupportsTrafficBasedKeepAlive": false,
      "SupportsExceedingPowerOfNonOperationalState": false,
      "Supports128BitHostId": false,
      "SupportsReservations": false
    }
  },
  "NVMeSMARTCriticalWarnings": {
    "MediaInReadOnly": false,
    "OverallSubsystemDegraded": false,
    "SpareCapacityWornOut": false
  },
  "MaxQueueSize": 1
},

"Links": {
```

```
"NetworkDeviceFunctions": [  
  {  
    "@odata.id": "/redfish/v1/Chassis/NVMeOpaqueArray/NetworkAdapters/OpaqueArrayNetworkAdapter/NetworkDeviceFunctions/11100"  
  },  
  {  
    "@odata.id": "/redfish/v1/Chassis/NVMeOpaqueArray/NetworkAdapters/OpaqueArrayNetworkAdapter/NetworkDeviceFunctions/11101"  
  },  
  {  
    "@odata.id": "/redfish/v1/Chassis/NVMeOpaqueArray/NetworkAdapters/OpaqueArrayNetworkAdapter/NetworkDeviceFunctions/11102"  
  },  
  {  
    "@odata.id": "/redfish/v1/Chassis/NVMeOpaqueArray/NetworkAdapters/OpaqueArrayNetworkAdapter/NetworkDeviceFunctions/11103"  
  }  
]  
,  
"@odata.id": "/redfish/v1/Storage/OpaqueArray/Controllers/NVMeAdminController",  
"@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."  
}
```

6.4.1.2 Property Mapping

6.4.1.2.1 Actions.#StorageController.RunSelfTest The mapping for `Actions.#StorageController.RunSelfTest` is summarized in Table 23.

Table 23: `Actions.#StorageController.RunSelfTest` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Actions.#StorageController.RunSelfTest</code>	NVM Spec Property / Field: Device Self-test Command NVM Spec: Section:Figure NVM Base Specification 2.0: Section 5.9
Type	Action (Special form of POST)	NVMe Administrative command
Description	This action is used to initiate a self-test against this controller. This action should return either the <code>ResourceSelfTestFailed</code> OR <code>ResourceSelfTestCompleted</code> event.	The Device Self-test command is used to start a device self-test operation or abort a device self-test operation
Long Description	This action is used to initiate a self-test against this controller. This action should return either the <code>ResourceSelfTestFailed</code> OR <code>ResourceSelfTestCompleted</code> event.	The Device Self-test command is used to start a device self-test operation or abort a device self-test operation (refer to NVMe 2.0 Base Specification section 8.6).
Mandatory	Optional	Optional
Notes		The Device Self-test command is used specifically to: a) start a short device self-test operation;b) start an extended device self-test operation;c) start a vendor specific device self-test operation; ord) abort a device self-test operation already in process.

6.4.1.2.2 Assembly The mapping for `Assembly` is summarized in Table 24.

Table 24: Assembly mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Description	The Assembly schema defines an assembly. Assembly information contains details about a device, such as part number, serial number, manufacturer, and production date. It also provides access to the original data for the assembly.	N/A
LongDescription	This Resource shall represent an assembly for a Redfish implementation. Assembly information contains details about a device, such as part number, serial number, manufacturer, and production date. It also provides access to the original data for the assembly.	N/A

Mandatory | Optional

Do Not Implement for NVMe drives, or devices with NVMe front ends, such as opaque arrays. ||| Notes |||

6.4.1.2.3 AssetTag The mapping for `AssetTag` is summarized in Table 25.

Table 25: Assembly mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	AssetTag	N/A
Type	Edm.String	N/A
Description	The user-assigned asset tag for this storage controller.	N/A
LongDescription	This property shall track the storage controller for inventory purposes.	N/A

Mandatory | Optional

Do Not Implement for NVMe Drives, or devices with NVMe front ends, such as opaque arrays. ||| Notes |||

6.4.1.2.4 CacheSummary The mapping for `CacheSummary` is summarized in Table 26.

Table 26: CacheSummary mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CacheSummary	N/A
Type	ComplexType	N/A
Description	The cache memory of the storage controller in general detail.	N/A
LongDescription	This property shall contain properties that describe the cache memory for this resource.	N/A
Mandatory	Do Not Implement	
Notes		This property exists for hw cache reporting in other RF/SF use cases. Not used in NVMe controllers.

6.4.1.2.5 ControllerRates The mapping for `ControllerRates` is summarized in Table 27.

Table 27: ControllerRates mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	ControllerRates	N/A
Type	ComplexType	N/A
Description	This property describes the various controller rates used for processes such as volume rebuild or consistency checks.	N/A
LongDescription	This object shall contain all the rate settings available on the controller.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.1.2.6 Description The mapping for `Description` is summarized in Table 28.

Table 28: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Optional	
Notes	In Redfish, Description is a read-only field.	Return the common description: “An NVM Admin Controller exposes capabilities that allow a host to manage an NVM subsystem. Admin controllers support commands providing management capabilities but does not provide IO access.”

6.4.1.2.7 FirmwareVersion The mapping for `FirmwareVersion` is summarized in Table 6.4.1.2.7.

Property	FirmwareVersion	NVM Spec Property / Field	Identify
Command	Identify Controller Data structure (CNS 01h)	Firmware Revision (FR)	

: FirmwareVersion mapping

Property | FirmwareVersion | **NVM Spec Property / Field:** Identify

Command / Identify Controller Data structure (CNS 01h): Firmware Revision (FR)

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1:Figure 275: Bytes 71:64 | | Type | String | String | | Description | The firmware version of this storage controller. | The currently active firmware revision for the domain of which this controller is a part. | | LongDescription | This property shall contain the firmware version as defined by the manufacturer for the associated storage controller. | The currently active firmware revision for the domain of which this controller is a part. | | Mandatory | Mandatory | Mandatory | | Notes | |

6.4.1.2.8 Identifiers The mapping for `Identifiers` is summarized in Table 29.

Table 29: Identifiers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers	N/A
Type	Collection(Resource.Identifier)	N/A
Description	The Durable names for the storage controller.	N/A
LongDescription	This property shall contain a list of all known durable names for the associated storage controller.	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.1.2.9 Identifiers.DurableName The mapping for `Identifiers.DurableName` is summarized in Table 30.

Table 30: Identifiers.DurableName mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableName	N/A
Type	Variable - see notes	N/A
Description	The world-wide, persistent name of the resource.	N/A
LongDescription	This property shall contain the world-wide unique identifier for the resource. The string shall be in the <code>Identifiers.DurableNameFormat</code> property value format.	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.1.2.10 Identifiers.DurableNameFormat The mapping for `Identifiers.DurableNameFormat` is summarized in Table 31.

Table 31: Identifiers.DurableNameFormat mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableNameFormat	N/A
Type	enum (DurableNameFormat)	N/A
Description	The format of the durable name property.	N/A
LongDescription	This property shall represent the format of the DurableName property.	N/A
Mandatory	Do Not Implement.	
Notes	There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.1.2.11 Links.AttachedVolumes The mapping for `Links.AttachedVolumes` is summarized in Table 32.

Table 32: `Links.AttachedVolumes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.AttachedVolumes</code>	N/A
Type	<code>Collection(Volume.Volume)</code>	N/A
Description	An array of links to volumes that are attached to this controller instance.	N/A
LongDescription	This property shall contain a link to the Resources of type <code>Volume</code> that are attached to this instance of storage controller.	N/A

Mandatory | Optional

Do Not Implement for admin controllers. ||| Notes |||

6.4.1.2.12 Links.NetworkDeviceFunctions The mapping for `Links.NetworkDeviceFunctions` is summarized in Table 35.

Table 33: `Links.NetworkDeviceFunctions` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.NetworkDeviceFunctions</code>	N/A
Type	<code>Collection(NetworkDeviceFunction.NetworkDeviceFunction)</code>	N/A
Description	The network device functions that provide connectivity to this controller.	N/A
LongDescription	This property shall contain an array of links to resources of type <code>NetworkDeviceFunction</code> that represent the devices that provide connectivity to this controller.	N/A
Mandatory	Optional. Recommended for NVMe-oF configurations and more complex devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.1.2.13 Location The mapping for `Location` is summarized in Table 34.

Table 34: Location mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Location	N/A
Type	Collection(Resource.Location)	N/A
Description	The location of the storage controller.	N/A
LongDescription	This property shall contain location information of the associated storage controller.	N/A

Mandatory | Optional

Do Not Implement for NVM Drives or more complex devices with an NVMe front end, such as opaque arrays. ||| Notes |||

6.4.1.2.14 Manufacturer The mapping for `Manufacturer` is summarized in Table 6.4.1.2.14.

Redfish/Swordfish	NVMe / NVMe-oF
Manufacturer	Manufacturer

: Manufacturer mapping

Property | Manufacturer | **NVM Spec Property / Field:** Identify

Command / Identify Controller Datastructure (CNS 01h) / PCI Vendor ID (VID)

NVM Spec: Section: Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 00:01 || Type | String | 16-bit number in little endian format. || Description | The manufacturer of this storage controller. | The company vendor identifier || LongDescription | This property shall contain the name of the organization responsible for producing the storage controller. This organization might be the entity from whom the storage controller is purchased, but this is not necessarily true. | The company vendor identifier that is assigned by the PCI SIG. This is the same value as reported in the ID register || Mandatory | Optional | NVMe: Mandatory

NVMe-oF: DNI || Notes | End clients expect to see the name of the company (e.g.; Contoso, BestVendor). While the value may be filled from the IdentifyController PCI Vendor ID, it would be preferable to have this filled with the actual string value of the company name. ||

6.4.1.2.15 Model The mapping for `Model` is summarized in Table 6.4.1.2.15.

```
+--+-----+-----+ || Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
+--+-----+-----+
```

: Model mapping

Property | Model | **NVM Spec Property / Field:** Identify Command /

Identify Controller Datastructure (CNS 01h) / Model Number (MN)

NVM Spec: Section: Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 24:63 || Type | String | String || Description | The model number for the storage controller. | Model Number (MN) || LongDescription | This property shall contain the name by which the manufacturer generally refers to the storage controller. | Contains the model number for the NVM subsystem that is assigned by the vendor as an ASCII string. Refer to NVMe 2.0 Section 7.10 for unique identifier requirements. Refer to NVMe 2.0 Section 1.5 for ASCII string requirements || Mandatory | Recommended | Mandatory || Notes || |

6.4.1.2.16 Name The mapping for `Name` is summarized in Table 6.4.1.2.16.

```
+--+-----+-----+ || Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
+--+-----+-----+
```

: Name mapping

Property | Name | **NVM Spec Property / Field:** Identify Command /

Identify Controller Data Data structure (CNS 01h) / Controller ID (CNTLID)

NVM Spec: Section: Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 78:79 || Type | String | 16-bit hex value || Description | The name of the resource or array member. | Controller ID || LongDescription | This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the ‘Name’ reserved word format. | The NVM subsystem unique controller identifier associated with the controller. || Mandatory | Mandatory | Mandatory || Notes | In Redfish, Name is a read-only field. | Map the CNTLID field to a string with the format: “0xABCD” |

6.4.1.2.17 NVMeControllerProperties.ControllerType The mapping for `NVMeControllerProperties.ControllerType` is summarized in Table 35.

Table 35: `NVMeControllerProperties.ControllerType` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.ControllerType</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Type (CNTRL_TYPE) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 111
Type	<code>StorageController.v1_0_0.NVMeControllerType</code>	Hex value
Description	This property specifies the type of NVMe Controller.	Controller Type
LongDescription	This property shall specify the type of NVMe Controller.	This field specifies the controller type. Values reported: 0h Reserved (controller type not reported)1h I/O controller2h Discovery controller3h Administrative controller
Mandatory	Mandatory	Mandatory
Notes	This property must be used to specify the type of NVMe Controller. For an admin controller, set to Admin.	For Admin Controller - value in Identify Controller is '03h'. Return "Admin"

6.4.1.2.18 NVMeControllerProperties.NVMeVersion The mapping for `NVMeControllerProperties.NVMeVersion` is summarized in Table 6.4.1.2.18.

Redfish/Swordfish	NVMe / NVMe-oF

: `NVMeControllerProperties.NVMeVersion` mapping

Property | `NVMeControllerProperties.NVMeVersion` | **NVM Spec Property /

Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / Version (VER)

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 80:83 || Type | String | 32-bit value. || Description | The version of the NVMe Base Specification supported. | Version (VER) || LongDescription | This property shall specify the type of NVMe Controller. | Indicates the major, minor, and tertiary version of the NVM Express base specification that the controller implementation supports.

bits 31:16 indicate the major version bits 15:08 indicate the minor version bits 07:00 indicate the tertiary version number || Mandatory | Optional | Mandatory || Notes || Implementations compliant to NVM Express Base Specification revision 1.2 or later shall report a non-zero value in this field. Valid versions of the specification are: 1.0, 1.1, 1.2, 1.2.1, 1.3, 1.4, and 2.0. |

6.4.1.2.19 NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity The mapping for `NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity` is summarized in Table 42.

```
+--+-----+-----+ | | Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
+--+-----+-----+
```

: `NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity` mapping

Property |

`NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity` | **NVM Spec Property / Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 07 (Namespace Granularity)

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 07 || Type | Boolean |Single bit (bool) || Description | Indicates whether or not the controller supports reporting of Namespace Granularity. | Indicates whether or not the controller supports reporting of Namespace Granularity. || LongDescription | This property shall indicate whether or not the controller supports reporting of Namespace Granularity. | This property shall indicate whether or not the controller supports reporting of Namespace Granularity. || Mandatory | Optional

Recommended for NVM Drives and more complex devices with NVMe front ends, such as opaque arrays. ||| Notes |||

6.4.1.2.20 NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations` is summarized in Table 6.4.1.2.20.

Property	Redfish/Swordfish	NVMe / NVMe-oF
<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations</code>		

: NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations Mapping

Property |

`NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations` | **NVM Spec Property / Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 08 (SQ Associations)

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 08 || Type | Boolean |Single bit (bool) || Description | Indicates whether or not the controller supports SQ Associations. | Indicates whether or not the controller supports SQ Associations. || LongDescription | This property shall indicate whether or not the controller supports SQ Associations. | This property shall indicate whether or not the controller supports SQ Associations. || Mandatory | Optional | Mandatory || Notes |||

6.4.1.2.21 NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive The mapping for NVMeControllerProperties.NVMeControllerAttributes.TrafficBasedKeepAlive is summarized in Table 6.4.1.2.21.

```
+--+-----+-----+ | | Redfish/Swordfish | NVMe / NVMe-oF | +:==+:-----+:-----+
+--+-----+-----+
```

: NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive mapping

Property |

NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive | **NVM Spec Property / Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 06 (Traffic Based Keep Alive Support – TBKAS)

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 06 | | Type | Boolean |Single bit (bool) | | Description | Indicates whether or not the controller supports restarting KeepAlive Timer if traffic is processed from an admin command or IO during KeepAlive Timeout interval. | Indicates if the the host and controller are allowed to restart the Traffic Based Keep Alive Timer in the presence of Admin or I/O command processing. | | LongDescription | This property shall indicate whether or not the controller supports restarting KeepAlive Timer if traffic is processed from an admin command or IO during KeepAlive Timeout interval. | This property shall indicate whether or not the the host and controller are allowed to restart the Traffic Based Keep Alive Timer in the presence of Admin or I/O command processing. | | Mandatory | Optional

Mandatory for Ethernet-Attach Drives;

Mandatory for more complex devices with NVMe front ends, such as opaque arrays. | Mandatory | | Notes | | |

6.4.1.2.22 NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState

The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState` is summarized in Table 6.4.1.2.22.

```
+--+-----+-----+ | Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
+--+-----+-----+
```

: `NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState` mapping

Property | `NVMeControllerProperties.NVMeControllerAttributes`.

`SupportsExceedingPowerOfNonOperationalState` | **NVM Spec Property / Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 01 (Non-Operational Power State Permissive Mode)

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 01 | Type | Boolean | Single bit (bool) | Description | Indicates whether or not the controller supports exceeding Power of NonOperational State in order to execute controller initiated background operations in a non-operational power state. | Indicates whether or not the controller supports host control of whether the controller may temporarily exceed the power of a non-operational power state for the purpose of executing controller initiated background operations in a non-operational power state (i.e., Non-Operational Power State Permissive Mode supported). | LongDescription | This property shall indicate whether or not the controller supports exceeding Power of NonOperational State in order to execute controller initiated background operations in a non-operational power state. | This property indicates whether or not the controller supports host control of whether the controller may temporarily exceed the power of a non-operational power state for the purpose of executing controller initiated background operations in a non-operational power state (i.e., Non-Operational Power State Permissive Mode supported). | Mandatory | Optional | Mandatory | Notes |

6.4.1.2.23 NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId The mapping for `NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId` is summarized in Table 6.4.1.2.23.

	Redfish/Swordfish	NVMe / NVMe-oF	
--	-------------------	----------------	--

: NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId mapping

Property |

NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId | **NVM Spec Property / Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 00

NVM Spec: Section:Figure

NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 00 || Type | Boolean | Single bit (bool) || Description | Indicates whether or not the controller supports a 128-bit Host Identifier. | Indicates whether or not the controller supports a 128-bit Host Identifier. || LongDescription | This property shall indicate whether or not the controller supports a 128-bit Host Identifier. | This property shall indicate whether or not the controller supports a 128-bit Host Identifier. || Mandatory | Mandatory | Mandatory || Notes |||

6.4.1.2.24 NVMeControllerProperties.MaxQueueSize The mapping for `NVMeControllerProperties.MaxQueueSize` is summarized in Table 6.4.1.2.24.

Redfish/Swordfish	NVMe / NVMe-oF
<code>NVMeControllerProperties.ANCharacteristics</code>	<code>NVMeControllerProperties.MaxQueueSize</code>

: `NVMeControllerProperties.ANCharacteristics` mapping

Property | `NVMeControllerProperties.MaxQueueSize` | **NVM Spec Property

/ Field:** Property Get Command / Controller Capabilities (CAP) - Offset 0h / Maximum Queue Entries Supported (MQES)

NVM Spec: Section:Figure

NVMe 2.0: Section 3.1.3.1: Figure 36 : Bits 15:00 | | Type | Int64 | 16-bit value | | Description | Indicates the maximum individual queue size that an NVMe IO Controller supports. | Indicates the maximum individual queue size that the controller supports. | | LongDescription | This property shall contain the maximum individual queue entry size supported per queue. This is a zero-based value, where the minimum value is one, indicating two entries. | Indicates the maximum individual queue size that the controller supports. This is a 0's based value. The minimum value is 1h, indicating two entries. | | Mandatory | Optional

Mandatory for more complex devices with NVMe front ends, such as opaque arrays. | Mandatory | | Notes | For PCIe, this applies to both submission and completion queues. For NVMe-oF, this applies to only submission queues. | For NVMe over PCIe implementations, this value applies to the I/O Submission Queues and I/O Completion Queues that the host creates.

For NVMe over Fabrics implementations, this value applies to only the I/O Submission Queues that the host creates. |

6.4.1.2.25 NVMeControllerProperties.ANCharacteristics The mapping for `NVMeControllerProperties.ANCharacteristics` is summarized in Table 36.

Table 36: `NNVMeControllerProperties.ANCharacteristics` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.ANCharacteristics</code>	NVM Spec Property / Field: Asymmetric Namespace Access Log (Log ID 0ch) NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.13: Figures 221 and 222
Type	Collection(StorageController.v1_0_0.ANCharacteristics)	Log Page.
Description	This property contains the combination of ANA type and volume information.	This log consists of a header describing the log and descriptors containing the asymmetric namespace access information for ANA Groups
LongDescription	This property shall contain the combination of ANA type and volume information.	This log consists of a header describing the log and descriptors containing the asymmetric namespace access information for ANA Groups that contain namespaces that are attached to the controller processing the command. ANA Group Descriptors shall be returned in ascending ANA Group Identifier order
Mandatory	Do Not Implement	Prohibited
Notes		

6.4.1.2.26 NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSystemDegraded The mapping for NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSystemDegraded is summarized in Table 6.4.1.2.26.

```
+--+-----+-----+ | | Redfish/Swordfish | NVMe / NVMe-oF | +:==+:-----+:-----+
+--+-----+-----+
```

: NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSystemDegraded mapping

Property |

NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSystemDegraded | **NVM Spec Property / Field:** SMART / Health Information Log (Log ID 02h) / Critical Warning / Subsystem Reliability has been degraded

NVM Spec: Section:Figure

NVMe 2.0: Section 5.16.1.3: Figure 207: Bytes 00, Bits 02 | | Type | Boolean | Single bit (bool) | | Description | Indicates that the NVM subsystem reliability has been compromised. | Indicates that the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability. | | LongDescription | This property shall indicate that the NVM subsystem reliability has been compromised. | This property Indicates that the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability. | | Mandatory | Mandatory | Optional | | Notes | |

6.4.1.2.27 NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed` is summarized in Table 50.

```
+--+-----+-----+ | | Redfish/Swordfish | NVMe / NVMe-oF | +:--+:-:-----+:-:-----+
+--+-----+-----+
```

: `NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed` mapping

Property |

`NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed` | **NVM Spec Property / Field:** SMART / Health Information Log (Log ID 02h) / Critical Warning / volatile memory backup device has failed

NVM Spec: Section:Figure

NVMe 2.0: Section 5.16.1.3: Figure 207: Bytes 00, Bits 04 | | Type | Boolean | Single bit (bool) | | Description | Indicates that the volatile memory backup device has failed. | Indicates that the volatile memory backup device has failed. | | LongDescription | This property shall indicate that the volatile memory backup device has failed. | This warning indicates that the volatile memory backup device has failed. | | Mandatory | Optional

Recommended for NVM Drives.

Required for more complex devices with NVMe front ends, such as opaque arrays. | Optional | | Notes | | This field is only valid if the controller has a volatile memory backup solution. |

6.4.1.2.28 SKU The mapping for `SKU` is summarized in Table 37.

Table 37: SKU mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SKU	N/A
Type	Edm.String	N/A
Description	The SKU for this storage controller.	N/A
LongDescription	This property shall contain the stock-keeping unit number for this storage storage controller.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.1.2.29 SpeedGbps The mapping for `SpeedGbps` is summarized in Table 38.

Table 38: SpeedGbps mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SpeedGbps	N/A
Type	Decimal	N/A
Description	The maximum speed of the storage controller's device interface.	N/A
LongDescription	This property shall represent the maximum supported speed of the storage bus interface, in Gbit/s. The specified interface connects the controller to the storage devices, not the controller to a host. For example, SAS bus not PCIe host bus.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.1.2.30 Status.Health The mapping for `Status.Health` is summarized in Table 6.4.1.2.30.

```
+--+-----+-----+ | | Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
+--+-----+-----+
```

: Status.Health mapping

Property | `Status.Health` | **NVM Spec Property / Field:** Property Get

command / Controller Status (CSTS): - Offset 1Ch

NVM Spec: Section:Figure

NVMe 2.0: Section 3.1.3.6: Figure 47

NVM Spec Property / Field: Get Log Page command / SMART/Health Information Log (Log Identifier 02h) **NVM Spec: Section:Figure** NVMe 2.0: Section 5.16.1.3: Figure 207 | | Type | Resource.Health | 32-bit value | | Description | The health state of this resource in the absence of its dependent resources. | This field indicates critical warnings for the state of the controller from both the SMART & general health information log and the Controller Status Property | | LongDescription | This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification. | This field indicates critical warnings for the state of the controller from the SMART and general health information . The information provided is over the life of the controller and is retained across power cycles. | | Mandatory | Mandatory | Optional | | Notes | Possible Values: OK / Warning / Critical | This comes from CSTS Controller Status.

Redfish/Swordfish “OK” corresponds to NVMe/NVMe-oF ‘Ready’ (RDY) bit set to ‘1’ and no warnings in the CSTS or SMART/Health information log page

Redfish/Swordfish “Warning” corresponds to “Volatile Memory Backup Device has failed”, “NVM subsystem reliability has been degraded”, “Temperature greater than or equal to an over temperature threshold”, “temperature less than or equal to an under temperature threshold”, or “available spare capacity has fallen below the threshold”

Redfish/Swordfish “Critical” corresponds to NVMe/NVMe-oF ‘Controller Fatal Status’ (CSTS.CFS) bit set to ‘1’ which indicates that a serious error condition has occurred. |

6.4.1.2.31 Status.State The mapping for `Status.State` is summarized in Table 39.

Table 39: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	NVM Spec Property / Field: Property Get Command / CSTS – Controller Status (Offset 1Ch) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.6: Figure 47
Type	Resource.State (enum)	32-bit value
Description	The known state of the resource, such as, enabled.	Controller Status
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	Controller Status
Mandatory	Mandatory	Mandatory

Notes | Possible values: Enabled / Disabled / StandbyOffline /

StandbySpare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified | Ready (CSTS.RDY) maps to Enabled, Shutdown (CSTS.SHST) value will tell you if shutdown is in progress or complete (StandbyOffline), Processing Paused (CSTS.PP) maps to Deferring.

If both Ready and Shutdown are indicated, then the system should indicate StandbyOffline.

If both Ready and Processing Paused are indicated, then the system should indicate Deferring. |

6.4.1.2.32 SupportedControllerProtocols The mapping for `SupportedControllerProtocols` is summarized in Table 40.

Table 40: `SupportedControllerProtocols` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>SupportedControllerProtocols</code>	N/A
Type	<code>Collection(Protocol.Protocol)</code>	N/A
Description	The supported set of protocols for communicating to this storage controller.	N/A
LongDescription	This property shall contain the supported set of protocols for communicating to this storage controller.	N/A
Mandatory	Mandatory	
Notes	This is an array of protocols supported by the <code>StorageController</code> . This can be set to values including, but not limited to, PCIe, RDMA, NVMe-oF, RoCE, RoCEv2, and InfiniBand.	

6.4.1.2.33 SupportedDeviceProtocols The mapping for SupportedDeviceProtocols is summarized in Table 41.

Table 41: SupportedDeviceProtocols mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SupportedDeviceProtocols	N/A
Type	Collection(Protocol.Protocol)	N/A
Description	The protocols that the storage controller can use to communicate with attached devices.	N/A
LongDescription	This property shall contain the set of protocols this storage controller can use to communicate with attached devices.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.2 Discovery Controller

6.4.2.1 Mockup The following mockup shows a sample representation of the StorageController schema used to represent a Discovery Controller.

```
{
  "@odata.id": "/redfish/v1/Storage/NVMeoF-Discovery/Controllers/NVMeDiscoveryController",
  "@odata.type": "#StorageController.v1_7_0.StorageController",
  "Id": "9",
  "Name": "NVMe Discovery Controller",
  "Description": "An NVM IO controller is a general-purpose controller that provides access to logical block data and metadata stored on an NVM
  ↵ subsystem's non-volatile storage medium. IO Controllers may also support management capabilities.",

  "Status": {
    "State": "Enabled",
    "Health": "OK"
  },
  "Id": "NVMeDiscoveryController",

  "SupportedControllerProtocols": [
    "TCP", "RoCEv2", "FC"
  ],
  "SupportedDeviceProtocols": [
    "NVMeOverFabrics"
  ],

  "NVMeControllerProperties": {
    "NVMeVersion": "1.3",
    "NVMeControllerAttributes": {
      "ReportsUUIDList": false,
      "SupportsSQAssociations": false,
      "ReportsNamespaceGranularity": false,
      "SupportsTrafficBasedKeepAlive": false,
      "SupportsPredictableLatencyMode": false,
      "SupportsEnduranceGroups": false,
      "SupportsReadRecoveryLevels": false,
      "SupportsNVMSets": false,
      "SupportsExceedingPowerOfNonOperationalState": false,
      "Supports128BitHostId": false
    }
  },
  "Links": {
    "Endpoints": [{
      "@odata.id": "/redfish/v1/Fabrics/NVMe-oF/Endpoints/NVMeSubsystemEndpoint1"
    }],
    "NVMeDiscoveredSubsystems": [{
      "@odata.id": "/redfish/v1/Storage/NVMeoF-SS1"
    }
  ]
}
```

```
  },  
  {  
    "@odata.id": "/redfish/v1/Storage/NVMeoF-SS2"  
  }  
]  
},  
"@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."  
}
```

6.4.2.2 Property Mapping

6.4.2.2.1 Assembly The mapping for `Assembly` is summarized in Table 42.

Table 42: Assembly mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Assembly	N/A
Type	ComplexType	N/A
Description	The Assembly schema defines an assembly. Assembly information contains details about a device, such as part number, serial number, manufacturer, and production date. It also provides access to the original data for the assembly.	N/A
LongDescription	This Resource shall represent an assembly for a Redfish implementation. Assembly information contains details about a device, such as part number, serial number, manufacturer, and production date. It also provides access to the original data for the assembly.	N/A
Mandatory	Optional Do Not Implement for NVMe drives, or devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.2.2.2 AssetTag The mapping for `AssetTag` is summarized in Table 43.

Table 43: AssetTag mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	AssetTag	N/A
Type	Edm.String	N/A
Description	The user-assigned asset tag for this storage controller.	N/A
LongDescription	This property shall track the storage controller for inventory purposes.	N/A
Mandatory	Optional Do Not Implement for NVMe Drives, or devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.2.2.3 CacheSummary

The mapping for `CacheSummary` is summarized in Table 44.

Table 44: CacheSummary mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CacheSummary	N/A
Type	ComplexType	N/A
Description	The cache memory of the storage controller in general detail.	N/A
LongDescription	This property shall contain properties that describe the cache memory for this resource.	N/A
Mandatory	Do Not Implement	
Notes		This property exists for hw cache reporting in other RF/SF use cases. Not used in NVMe controllers.

6.4.2.2.4 ControllerRates The mapping for `ControllerRates` is summarized in Table 45.

Table 45: ControllerRates mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	ControllerRates	N/A
Type	ComplexType	N/A
Description	This property describes the various controller rates used for processes such as volume rebuild or consistency checks.	N/A
LongDescription	This object shall contain all the rate settings available on the controller.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.2.2.5 Description The mapping for `Description` is summarized in Table 46.

Table 46: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Optional	
Notes	In Redfish, <code>Description</code> is a read-only field.	Return the common description: “An NVM Discovery Controller exposes capabilities that allow a host to retrieve information required to connect to one or more NVM Subsystems. Discovery controllers only support commands providing discovery capabilities; they do not provide IO or management access.”

6.4.2.2.6 FirmwareVersion

The mapping for `FirmwareVersion` is summarized in Table 47.

Table 47: FirmwareVersion mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	FirmwareVersion	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h): Firmware Revision (FR) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1:Figure 275: Bytes 71:64
Type	String	String
Description	The firmware version of this storage controller.	The currently active firmware revision for the domain of which this controller is a part.
LongDescription	This property shall contain the firmware version as defined by the manufacturer for the associated storage controller.	The currently active firmware revision for the domain of which this controller is a part.
Mandatory	Mandatory	Mandatory
Notes		Return the currently active firmware revision information.

6.4.2.2.7 Identifiers The mapping for `Identifiers` is summarized in Table 48.

Table 48: Identifiers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers	N/A
Type	Collection(Resource.Identifier)	N/A
Description	The Durable names for the storage controller.	N/A
LongDescription	This property shall contain a list of all known durable names for the associated storage controller.	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.2.2.8 Identifiers.DurableName

The mapping for `Identifiers.DurableName` is summarized in Table 49.

Table 49: Identifiers.DurableName mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableName	N/A
Type	Variable - see notes	N/A
Description	The world-wide, persistent name of the resource.	N/A
LongDescription	This property shall contain the world-wide unique identifier for the resource. The string shall be in the <code>Identifiers.DurableNameFormat</code> property value format.	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.2.2.9 Identifiers.DurableNameFormat The mapping for `Identifiers.DurableNameFormat` is summarized in Table 50.

Table 50: Identifiers.DurableNameFormat mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableNameFormat	N/A
Type	enum (DurableNameFormat)	N/A
Description	The Durable names for the storage controller.	N/A
LongDescription	This property shall contain a list of the types for all known durable names for the associated storage controller. The type determines the length of the corresponding Namespace ID	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.2.2.10 Links.AttachedVolumes The mapping for `Links.AttachedVolumes` is summarized in Table 51.

Table 51: `Links.AttachedVolumes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.AttachedVolumes</code>	N/A
Type	<code>Collection(Volume.Volume)</code>	N/A
Description	An array of links to volumes that are attached to this controller instance.	N/A
LongDescription	This property shall contain a link to the Resources of type Volume that are attached to this instance of storage controller.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.2.2.11 Links.Endpoints The mapping for `Links.Endpoints` is summarized in Table 52.

Table 52: Links.Endpoints mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.Endpoints	N/A
Type	Collection(Endpoint.Endpoint)	N/A
Description	An array of links to the endpoints that connect to this controller.	N/A
LongDescription	This property shall contain an array of links to the Resources of type Endpoint associated with this controller.	N/A
Mandatory	Mandatory	
Notes		

6.4.2.2.12 Links.Connections

The mapping for `Links.Connections` is summarized in Table 53.

Table 53: `Links.Connections` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.Connections</code>	N/A
Type	<code>Collection(Connection.Connection)</code>	N/A
Description	An array of links to volumes that are attached to this controller instance.	N/A
LongDescription	This property shall contain a link to the Resources of type Volume that are attached to this instance of storage controller.	N/A
Mandatory	Optional	
Notes	This contains the information used to represent the allowed hosts.	This property contains pointers to the Connections objects. The information about allowed hosts is mapped to the Connections objects for NVMe-oF configurations.

6.4.2.2.13 Links.NetworkDeviceFunctions The mapping for `Links.NetworkDeviceFunctions` is summarized in Table 54.

Table 54: `Links.NetworkDeviceFunctions` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.NetworkDeviceFunctions</code>	N/A
Type	<code>Collection(NetworkDeviceFunction.NetworkDeviceFunction)</code>	N/A
Description	The network device functions that provide connectivity to this controller.	N/A
LongDescription	This property shall contain an array of links to resources of type <code>NetworkDeviceFunction</code> that represent the devices that provide connectivity to this controller.	N/A
Mandatory	Optional Recommended to implement for more complex devices with NVMe front ends, such as opaque arrays.	
Notes	For NVMe-oF configurations.	

6.4.2.2.14 Links.NVMeDiscoveredSubsystems The mapping for `Links.NVMeDiscoveredSubsystems` is summarized in Table 55.

Table 55: Links.NVMeDiscoveredSubsystems mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.NVMeDiscoveredSubsystems	NVM Spec Property / Field: NVMe 2.0 Base Specification: 5.16.1.23 Discovery Log Page (Log Identifier 70h) GetLogPage
Type	Collection(Storage.Storage)	N/A
Description	The NVMe subsystems discovered by this discovery controller.	The Discovery Log Page provides an inventory of NVM subsystems with which a host may attempt to form an association. The Discovery Log Page may be specific to the host requesting the log.
LongDescription	This property shall contain an array of links to resources of type Storage that represent the NVMe subsystems discovered by this discovery controller. This property shall only be present if ControllerType in NVMeControllerProperties contains <code>Discovery</code> .	
Mandatory	Mandatory	Mandatory
Notes	This contains the list of all Subsystems discovered by this Discovery Controller.	The Discovery Log Page is persistent across power cycles.

6.4.2.2.15 Location The mapping for `Location` is summarized in Table 56.

Table 56: Location mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Location	N/A
Type	Collection(Resource.Location)	N/A
Description	The location of the storage controller.	N/A
LongDescription	This property shall contain location information of the associated storage controller.	N/A
Mandatory	Optional Do Not Implement for NVM Drives, or more complex devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.2.2.16 Manufacturer The mapping for `Manufacturer` is summarized in Table 57.

Table 57: Manufacturer mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Manufacturer	N/A
Type	String	N/A
Description	The manufacturer of this storage controller.	N/A
LongDescription	This property shall contain the name of the organization responsible for producing the storage controller. This organization might be the entity from whom the storage controller is purchased, but this is not necessarily true.	N/A
Mandatory	Optional	Reserved (DNI)
Notes	End clients expect to see the name of the company (e.g., Contoso, BestVendor). While the value may be filled from the <code>IdentifyController PCI Vendor ID</code> or <code>SubsystemID</code> field, it would be preferable to have this filled with the actual string value of the company name.	

6.4.2.2.17 Model The mapping for `Model` is summarized in Table 58.

Table 58: Model mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Model	N/A
Type	String	N/A
Description	The model number for the storage controller.	N/A
LongDescription	This property shall contain the name by which the manufacturer generally refers to the storage controller.	N/A
Mandatory	Do Not Implement	Reserved (DNI) for Discovery Controllers
Notes		

6.4.2.2.18 Name The mapping for `Name` is summarized in Table 59.

Table 59: Name mapping

		Redfish/Swordfish
Property	Name	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller ID (CNTLID) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 78:79
Type	String	16-bit hex value
Description	The name of the resource or array member.	Controller ID
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	The NVM subsystem unique controller identifier associated with the controller.
Mandatory	Mandatory	Mandatory
Notes	In Redfish, Name is a read-only field.	Map the CNTLID field to a string with the format: "0xABCD"

6.4.2.2.19 NVMeControllerProperties.ControllerType The mapping for `NVMeControllerProperties.ControllerType` is summarized in Table 60.

Table 60: `NVMeControllerProperties.ControllerType` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.ControllerType</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Type (CNTRL_TYPE) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 111
Type	StorageController. <code>v1_0_0.NVMeControllerType</code>	Hex value
Description	This property specifies the type of NVMe Controller.	Controller Type
LongDescription	This property shall specify the type of NVMe Controller.	This field specifies the controller type. Values reported: 0h Reserved (controller type not reported)1h I/O controller2h Discovery controller3h Administrative controller
Mandatory	Mandatory	Mandatory
Notes	This property must be used to specify the type of NVMe Controller. For a discovery controller, set to Discovery.	For Discovery Controller - value in Identify Controller is '02h'. Return "Discovery"

6.4.2.2.20 NVMeControllerProperties.NVMeVersion The mapping for `NVMeControllerProperties.NVMeVersion` is summarized in Table 61.

Table 61: `NVMeControllerProperties.NVMeVersion` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeVersion</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Version (VER) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 80:83
Type	String	32-bit value.
Description	The version of the NVMe Base Specification supported.	Version (VER)
LongDescription	This property shall contain the version of the NVMe Base Specification supported.	Indicates the major, minor, and tertiary version of the NVM Express base specification that the controller implementation supports.
Mandatory	Recommended Mandatory if NVMe version is 1.2 or later.	Mandatory
Notes		Implementations compliant to NVM Express Base Specification revision 1.2 or later shall report a non-zero value in this field. Valid versions of the specification are: 1.0, 1.1, 1.2, 1.2.1, 1.3, 1.4, and 2.0. bits 31:16 indicate the major version bits 15:08 indicate the minor version bits 07:00 indicate the tertiary version number

6.4.2.2.21 NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive The mapping for `NVMeControllerProperties.NVMeControllerAttributes.TrafficBasedKeepAlive` is summarized in Table 62.

Table 62: `NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive` SupportsTrafficBasedKeepAlive mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive</code>	N/A
Type	Boolean	N/A
Description	Indicates whether or not the controller supports restarting KeepAlive Timer if traffic is processed from an admin command or IO during KeepAlive Timeout interval.	N/A
LongDescription	This property shall indicate whether or not the controller supports restarting KeepAlive Timer if traffic is processed from an admin command or IO during KeepAlive Timeout interval.	N/A
Mandatory	Do Not Implement	DNI (Reserved)
Notes		

6.4.2.2.22 NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState

The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState` is summarized in Table 63.

Table 63: `NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState</code>	N/A
Type	Boolean	N/A
Description	Indicates whether or not the controller supports exceeding Power of NonOperational State in order to execute controller initiated background operations in a non-operational power state.	N/A
LongDescription	This property shall indicate whether or not the controller supports exceeding Power of NonOperational State in order to execute controller initiated background operations in a non-operational power state.	N/A
Mandatory	Optional	DNI (Reserved)
Notes		

6.4.2.2.23 NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId The mapping for `NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId` is summarized in Table 64.

Table 64: `NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId` `Supports128BitHostId` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId</code>	N/A
Type	Boolean	N/A
Description	Indicates whether or not the controller supports a 128-bit Host Identifier.	N/A
LongDescription	This property shall indicate whether or not the controller supports a 128-bit Host Identifier.	N/A
Mandatory	Do Not Implement	DNI (Reserved)
Notes		

6.4.2.2.24 NVMeControllerProperties.MaxQueueSize The mapping for `NVMeControllerProperties.MaxQueueSize` is summarized in Table 65.

Table 65: `NVMeControllerProperties.MaxQueueSize` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.MaxQueueSize</code>	NVM Spec Property / Field: Property Get command / Controller Capabilities (CAP) - Offset 0h: Maximum Queue Entries Supported (MQES) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.1: Figure 36: Bits 15:00
Type	Int64	16-bit value
Description	Indicates the maximum individual queue size that an NVMe IO Controller supports.	Indicates the maximum individual queue size that the controller supports.
LongDescription	This property shall contain the maximum individual queue entry size supported per queue. This is a zero-based value, where the minimum value is one, indicating two entries. For PCIe, this applies to both submission and completion queues. For NVMe-oF, this applies to only submission queues.	indicates the maximum individual queue size that the controller supports. For NVMe over PCIe implementations, this value applies to the I/O Submission Queues and I/O Completion Queues that the host creates. For NVMe over Fabrics implementations, this value applies to only the I/O Submission Queues that the host creates. This is a 0's based value. The minimum value is 1h, indicating two entries.
Mandatory	Optional Mandatory for more complex devices with NVMe front ends, such as opaque arrays.	Mandatory
Notes		

6.4.2.2.25 NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemDegraded The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemDegraded` is summarized in Table 66.

Table 66: `NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemsDegraded` `OverallSubsystemDegraded` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemDegraded</code>	N/A
Type	Boolean	N/A
Description	Indicates that the NVM subsystem reliability has been compromised.	N/A
LongDescription	This property shall indicate that the NVM subsystem reliability has been compromised.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.2.2.26 NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut` is summarized in Table 67.

Table 67: `NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut` SpareCapacityWornOut mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut</code>	N/A
Type	Boolean	N/A
Description	Indicates that the available spare capacity has fallen below the threshold.	N/A
LongDescription	This property shall indicate that the available spare capacity has fallen below the threshold.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.2.2.27 NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed` is summarized in Table ??.

Table 68: `NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed</code>	N/A
Type	Boolean	N/A
Description	Indicates that the volatile memory backup device has failed.	N/A
LongDescription	This property shall indicate that the volatile memory backup device has failed.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.2.2.28 Status.Health The mapping for `Status.Health` is summarized in Table 69.

Table 69: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	N/A
Type	Resource.Health	N/A
Description	The health state of this resource in the absence of its dependent resources.	N/A
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	N/A
Mandatory	Mandatory	
Notes	Possible Values: OK / Warning / Critical	

6.4.2.2.29 Status.State

The mapping for `Status.State` is summarized in Table 70.

Table 70: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	NVM Spec Property / Field: Property Get Command / CSTS – Controller Status (Offset 1Ch) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.6: Figure 47
Type	Resource.State (enum)	32-bit value
Description	The known state of the resource, such as, enabled.	Controller Status
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	Controller Status
Mandatory	Mandatory	Mandatory
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySpare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified	Ready (CSTS.RDY) maps to Enabled Shutdown (CSTS.SHST) value will tell you if shutdown is in progress or complete (StandbyOffline) ProcessingPaused (CSTS.PP) maps to Deferring. If both Ready and Shutdown are indicated, then the system should indicate StandbyOffline. If both Ready and ProcessingPaused are indicated, then the system should indicate Deferring.

6.4.2.2.30 SupportedControllerProtocols The mapping for `SupportedControllerProtocols` is summarized in Table 71.

Table 71: `SupportedControllerProtocols` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>SupportedControllerProtocols</code>	N/A
Type	<code>Collection(Protocol.Protocol)</code>	N/A
Description	The supported set of protocols for communicating to this storage controller.	N/A
LongDescription	This property shall contain the supported set of protocols for communicating to this storage controller.	N/A
Mandatory	Mandatory	
Notes	This is an array of protocols supported by the <code>StorageController</code> . This can be set to values including, but not limited to, PCIe, RDMA, NVMe-oF, RoCE, RoCEv2, and InfiniBand.	

6.4.2.2.31 SupportedDeviceProtocols The mapping for SupportedDeviceProtocols is summarized in Table 72.

Table 72: SupportedDeviceProtocols mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SupportedDeviceProtocols	N/A
Type	Collection(Protocol.Protocol)	N/A
Description	The protocols that the storage controller can use to communicate with attached devices.	N/A
LongDescription	This property shall contain the set of protocols this storage controller can use to communicate with attached devices.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.3 IO Controller

6.4.3.1 Mockup The following mockup shows a sample representation of the StorageController schema used to represent an IO Controller.

```
{
  "@odata.type": "#StorageController.v1_7_0.StorageController",
  "Id": "NVMeIOController",
  "Name": "NVMe I/O Controller",
  "Description": "An NVMe IO controller is a general-purpose controller that provides access to logical block data and metadata stored on an NVMe
  ↵ subsystem's non-volatile storage medium. IO Controllers may also support management capabilities.",
  "Status": {
    "State": "Enabled",
    "Health": "OK"
  },
  "Manufacturer": "Best NVMe Vendor",
  "Model": "Simple NVMe Device",
  "SerialNumber": "NVME123456",
  "PartNumber": "NVM44",
  "FirmwareVersion": "1.0.0",
  "SupportedControllerProtocols": [
    "PCIe"
  ],
  "SupportedDeviceProtocols": [
    "NVMe"
  ],
  "SupportedRAIDTypes": [
    "None"
  ],
  "NVMeControllerProperties": {
    "ControllerType": "IO",
    "NVMeVersion": "1.3",
    "NVMeControllerAttributes": {
      "ReportsUUIDList": false,
      "SupportsSQAssociations": false,
      "ReportsNamespaceGranularity": false,
      "SupportsTrafficBasedKeepAlive": false,
      "SupportsPredictableLatencyMode": false,
      "SupportsEnduranceGroups": false,
      "SupportsReadRecoveryLevels": false,
      "SupportsNVMSets": false,
      "SupportsExceedingPowerOfNonOperationalState": false,
      "Supports128BitHostId": false
    }
  },
  "NVMeSMARTCriticalWarnings": {
    "MediaInReadOnly": false,
    "OverallSubsystemDegraded": false,
  }
}
```

```
    "SpareCapacityWornOut": false
  },
  "MaxQueueSize": 1
},
"Links": {
  "AttachedVolumes": [{
    "@odata.id": "/redfish/v1/Systems/Sys-1/Storage/SimplestNVMeSSD/Volumes/SimpleNamespace"
  }]
},
"Actions": {
  "#StorageController.SecuritySend": {
    "target": "/redfish/v1/Systems/Sys-1/Storage/SimplestNVMeSSD/Controllers/NVMeIOController/Actions/StorageController.SecuritySend"
  },
  "#StorageController.SecurityReceive": {
    "target": "/redfish/v1/Systems/Sys-1/Storage/SimplestNVMeSSD/Controllers/NVMeIOController/Actions/StorageController.SecurityReceive"
  }
},
"@odata.id": "/redfish/v1/Systems/Sys-1/Storage/SimplestNVMeSSD/Controllers/NVMeIOController",
"@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."
}
```

6.4.3.2 Property Mapping

6.4.3.2.1 Actions.#StorageController.RunSelfTest The mapping for `Actions.#StorageController.RunSelfTest` is summarized in Table 73.

Table 73: `Actions.#StorageController.RunSelfTest` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Actions.#StorageController.RunSelfTest</code>	NVM Spec Property / Field: Device Self-test Command NVM Spec: Section:Figure NVM Base Specification 2.0: 5.9 Device Self-test Command
Type	Action (Special form of POST)	NVMe Administrative command
Description	This action is used to initiate a self-test against this controller. This action should return either the <code>ResourceSelfTestFailed</code> OR <code>ResourceSelfTestCompleted</code> event.	The Device Self-test command is used to start a device self-test operation or abort a device self-test operation
Long Description	This action is used to initiate a self-test against this controller. This action should return either the <code>ResourceSelfTestFailed</code> OR <code>ResourceSelfTestCompleted</code> event.	The Device Self-test command is used to start a device self-test operation or abort a device self-test operation (refer to NVMe 2.0 Base Specification section 8.6).
Mandatory	Optional	Optional
Notes		The Device Self-test command is used specifically to: a) start a short device self-test operation;b) start an extended device self-test operation;c) start a vendor specific device self-test operation; ord) abort a device self-test operation already in process.

The mapping for `Assembly` is summarized in Table 74.

Table 74: `Assembly` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Assembly</code>	N/A
Type	<code>ComplexType</code>	N/A

	Redfish/Swordfish	NVMe / NVMe-oF
Des cription	The Assembly schema defines an assembly. Assembly information contains details about a device, such as part number, serial number, manufacturer, and production date. It also provides access to the original data for the assembly.	N/A
LongDes cription	This Resource shall represent an assembly for a Redfish implementation. Assembly information contains details about a device, such as part number, serial number, manufacturer, and production date. It also provides access to the original data for the assembly.	N/A
M andatory	Optional Do Not Implement for NVMe drives, or devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.3.2.2 AssetTag The mapping for `AssetTag` is summarized in Table 75.

Table 75: Assembly mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	AssetTag	N/A
Type	Edm.String	N/A
Description	The user-assigned asset tag for this storage controller.	N/A
LongDescription	This property shall track the storage controller for inventory purposes.	N/A
Mandatory	Optional Do Not Implement for NVMe Drives, or devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.3.2.3 CacheSummary The mapping for `CacheSummary` is summarized in Table 76.

Table 76: CacheSummary mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CacheSummary	N/A
Type	ComplexType	N/A
Description	The cache memory of the storage controller in general detail.	N/A
LongDescription	This property shall contain properties that describe the cache memory for this resource.	N/A
Mandatory	Do Not Implement	
Notes		This property exists for hw cache reporting in other RF/SF use cases. Not used in NVMe controllers.

6.4.3.2.4 ControllerRates The mapping for `ControllerRates` is summarized in Table 77.

Table 77: ControllerRates mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	ControllerRates	N/A
Type	ComplexType	N/A
Description	This property describes the various controller rates used for processes such as volume rebuild or consistency checks.	N/A
LongDescription	This object shall contain all the rate settings available on the controller.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.3.2.5 Description The mapping for `Description` is summarized in Table 78.

Table 78: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Description is a read-only field.	Return the common description: “An NVM IO controller is a general-purpose controller that provides access to logical block data and metadata stored on an NVM subsystem’s non-volatile storage medium. IO Controllers may also support management capabilities.”

6.4.3.2.6 FirmwareVersion The mapping for `FirmwareVersion` is summarized in Table 79.

Table 79: FirmwareVersion mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	FirmwareVersion	NVM Spec Property / Field: Identify Command / Identify Controller Data structure (CNS 01h) / Firmware Revision (FR) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1:Figure 275: Bytes 64 - 71
Type	String	String
Description	The firmware version of this storage controller.	The currently active firmware revision for the domain of which this controller is a part.
LongDescription	This property shall contain the firmware version as defined by the manufacturer for the associated storage controller.	The currently active firmware revision for the domain of which this controller is a part.

	Redfish/Swordfish	NVMe / NVMe-oF
Mandatory	Mandatory	Mandatory
Notes		Return the currently active firmware revision information.

6.4.3.2.7 Identifiers The mapping for `Identifiers` is summarized in Table 80.

Table 80: Identifiers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers	N/A
Description	The Durable names for the storage controller.	N/A
LongDescription	This property shall contain a list of all known durable names for the associated storage controller.	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.3.2.8 Identifiers.DurableName The mapping for `Identifiers.DurableName` is summarized in Table 81.

Table 81: Identifiers.DurableName mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableName	N/A
Type	Variable - see notes	N/A
Description	The Durable names for the storage controller.	N/A
LongDescription	The Durable names for the storage controller.	N/A
Mandatory	Optional	

	Redfish/Swordfish	NVMe / NVMe-oF
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.3.2.9 Identifiers.DurableNameFormat The mapping for `Identifiers.DurableNameFormat` is summarized in Table 82.

Table 82: Identifiers.DurableNameFormat mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableNameFormat	N/A
Type	enum (DurableNameFormat)	N/A
Description	The Durable names for the storage controller.	N/A
LongDescription	This property shall contain a list of the types for all known durable names for the associated storage controller. The type determines the length of the corresponding Namespace ID	N/A
Mandatory	Optional	
Notes	Recommend not implementing. There isn't a good mapping for these in the NVMe spec to a property that has an appropriate / mapping to a durable name format.	

6.4.3.2.10 Links.AttachedVolumes

The mapping for `Links.AttachedVolumes` is summarized in Table 83.

Table 83: `Links.AttachedVolumes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.AttachedVolumes</code>	NVM Spec Property / Field: Identify Command / Active Namespace ID list (CNS 07h) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.7
Type	Collection(<code>Volume.Volume</code>)	Namespace List
Description	An array of links to volumes that are attached to this controller instance.	A list of 1,024 namespace IDs is returned to the host containing active NSIDs in increasing order that are greater than the value specified in the Namespace Identifier (NSID) field of the command.
LongDescription	This property shall contain a link to the Resources of type <code>Volume</code> that are attached to this instance of storage controller.	A list of 1,024 namespace IDs is returned to the host containing active NSIDs in increasing order that are greater than the value specified in the Namespace Identifier (NSID) field of the command. The NSID field may be cleared to 0h to retrieve a Namespace List including the namespace starting with NSID of 1h. Reference NVMe 2.0: Section 4.4.2: Figure 135 (Namespace List Format)
Mandatory	Mandatory	Required
Notes	This contains a pointer to the set of namespaces attached to this IO Controller.	

6.4.3.2.11 Links.Endpoints The mapping for `Links.Endpoints` is summarized in Table 84.

Table 84: Links.Endpoints mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.Endpoints	N/A
Type	Collection(Endpoint.Endpoint)	N/A
Description	An array of links to the endpoints that connect to this controller.	N/A
LongDescription	This property shall contain an array of links to the Resources of type Endpoint associated with this controller.	N/A
Mandatory	Optional	
Notes	For NVMe-oF configurations.	

6.4.3.2.12 Links.Connections

The mapping for `Links.Connections` is summarized in Table 86.

Table 85: `Links.Connections` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.Connections</code>	N/A
Type	<code>Collection(Connection.Connection)</code>	N/A
Description	An array of links to volumes that are attached to this controller instance.	N/A
LongDescription	This property shall contain a link to the Resources of type <code>Volume</code> that are attached to this instance of storage controller.	N/A
Mandatory	Optional Do Not Implement for NVMe Drives.	
Notes	This contains the information used to represent the allowed hosts.	This property contains pointers to the <code>Connections</code> objects. The information about allowed hosts is mapped to the <code>Connections</code> objects for NVMe-oF configurations.

6.4.3.2.13 Links.NetworkDeviceFunctions

Table 86: `Links.NetworkDeviceFunctions` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.NetworkDeviceFunctions</code>	N/A
Type	<code>Collection(NetworkDeviceFunction.NetworkDeviceFunction)</code>	N/A
Description	The network device functions that provide connectivity to this controller.	N/A
LongDescription	This property shall contain an array of links to resources of type <code>NetworkDeviceFunction</code> that represent the devices that provide connectivity to this controller.	N/A
Mandatory	Optional Recommended for more complex devices with NVMe front ends, such as opaque arrays.	
Notes	For NVMe-oF configurations.	

6.4.3.2.14 Location The mapping for `Location` is summarized in Table 87.

Table 87: Location mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Location	N/A
Type	Collection(Resource.Location)	N/A
Description	The location of the storage controller.	N/A
LongDescription	This property shall contain location information of the associated storage controller.	N/A
Mandatory	Optional Do Not Implement for NVM Drives, or more complex devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.3.2.15 Manufacturer The mapping for `Manufacturer` is summarized in Table 88.

Table 88: Manufacturer mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Manufacturer	NVM Spec Property / Field:** Identify Command / Identify Controller Datastructure (CNS 01h) / PCI Vendor ID (VID) NVM Spec: Section: Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 00:01
Type	String	16-bit number in little endian format.
Description	The manufacturer of this storage controller.	The company vendor identifier
LongDescription	This property shall contain the name of the organization responsible for producing the storage controller. This organization might be the entity from whom the storage controller is purchased, but this is not necessarily true.	The company vendor identifier that is assigned by the PCI SIG. This is the same value as reported in the ID register
Mandatory	Optional	NVMe: Mandatory NVMe-oF: DNI

	Redfish/Swordfish	NVMe / NVMe-oF
Notes		End clients expect to see the name of the company (e.g.; Contoso, BestVendor). While the value may be filled from the IdentifyController PCI Vendor ID, it would be preferable to have this filled with the actual string value of the company name.

6.4.3.2.16 Model The mapping for `Model` is summarized in Table 89.

Table 89: Model mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Model	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Model Number (MN) NVM Spec: Section: Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 24:63
Type	String	String
Description	The model number for the storage controller.	Model Number (MN)
LongDescription	This property shall contain the name by which the manufacturer generally refers to the storage controller.	Contains the model number for the NVM subsystem that is assigned by the vendor as an ASCII string. Refer to section 7.10 for unique identifier requirements. Refer to section 1.5 for ASCII string requirements
Mandatory	Recommended	Mandatory
Notes		

6.4.3.2.17 Name The mapping for `Name` is summarized in Table 90.

Table 90: Name mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Name	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller ID (CNTLID) NVM Spec Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 78:79
Type	String	16-bit hex value
Description	The name of the resource or array member.	Controller ID
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	The NVM subsystem unique controller identifier associated with the controller.
Mandatory	Mandatory	Mandatory
Notes	In Redfish, Name is a read-only field.	Map the CNTLID field to a string with the format: "0xABCD"

6.4.3.2.18 NVMeControllerProperties.ControllerType The mapping for `NVMeControllerProperties.ControllerType` is summarized in Table 91.

Table 91: NVMeControllerProperties.ControllerType mapping

Redfish/Swordfish	NVMe / NVMe-oF	
Property	NVMeControllerProperties.ControllerType	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Type (CNTRL_TYPE) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 111
Type	StorageController.v1_0_0.NVMeControllerType	Hex value
Description	This property specifies the type of NVMe Controller.	Controller Type
LongDescription	This property shall specify the type of NVMe Controller.	This field specifies the controller type. Supported values: 0h Reserved (controller type not reported)1h I/O controller2h Discovery controller3h Administrative controller
Mandatory	Mandatory	Mandatory
Notes	This property must be used to specify the type of NVMe Controller. For an IO controller, set to IO.	For IO Controller - value in Identify Controller is '01h'. Return "IO"

6.4.3.2.19 NVMeControllerProperties.NVMeVersion The mapping for `NVMeControllerProperties.NVMeVersion` is summarized in Table 92.

Table 92: `NVMeControllerProperties.NVMeVersion` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeVersion</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Version (VER) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 80:83
Type	String	32-bit value. bits 31:16 indicate the major version bits 15:08 indicate the minor version bits 07:00 indicate the tertiary version number
Description	The version of the NVMe Base Specification supported.	Version (VER)
LongDescription	This property shall specify the type of NVMe Controller.	Indicates the major, minor, and tertiary version of the NVM Express base specification that the controller implementation supports.
Mandatory	Optional	Mandatory
Notes		Implementations compliant to NVM Express Base Specification revision 1.2 or later shall report a non-zero value in this field. Valid versions of the specification are: 1.0, 1.1, 1.2, 1.2.1, 1.3, 1.4, and 2.0.

6.4.3.2.20 NVMeControllerProperties.NVMeControllerAttributes.ReportsUUIDList The mapping for `NVMeControllerProperties.NVMeControllerAttributes.ReportsUUIDList` is summarized in Table 93.

Table 93: `NVMeControllerProperties.NVMeControllerAttributes.ReportsUUIDList` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.ReportsUUIDList</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT) : UUID List (Bit 9) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 99:96 (CTRATT): Bit 9 (UUID List)
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports reporting of a UUID list.	Indicated whether or not the controller supports reporting of a UUID List.
LongDescription	This property shall indicate whether or not the controller supports reporting of a UUID list.	Indicated whether or not the controller supports reporting of a UUID List.
Mandatory	Optional	Mandatory for IO Controllers
Notes		

6.4.3.2.21 NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations` is summarized in Table 94.

Table 94: SupportsSQAssociations mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeControllerProperties.NVMeControllerAttributes.SupportsSQAssociations	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure / Controller Attributes (CTRATT): Bit 08 (SQ Associations) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 08 (SQ Associations)
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports SQ Associations.	Indicates whether or not the controller supports SQ Associations.
LongDescription	This property shall indicate whether or not the controller supports SQ Associations.	This property shall indicate whether or not the controller supports SQ Associations.
Mandatory	Optional	Mandatory
Notes		

6.4.3.2.22 NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity The mapping for `NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity` is summarized in Table 95.

Table 95: `NVMeControllerProperties.NVMeControllerAttributes`.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.ReportsNamespaceGranularity</code>	NVM Spec Property / Field: Identify Comand / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 07 (Namespace Granularity) NVM Spec Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 07
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports reporting of Namespace Granularity.	Indicates whether or not the controller supports reporting of Namespace Granularity.
LongDescription	This property shall indicate whether or not the controller supports reporting of Namespace Granularity.	This property shall indicate whether or not the controller supports reporting of Namespace Granularity.
Mandatory	Optional Recommended for NVM Drives and more complex devices with NVMe front ends, such as opaque arrays.	Mandatory
Notes		

6.4.3.2.23 NVMeControllerProperties.NVMeControllerAttributes.SupportsReservations The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsReservations` is summarized in Table 96.

Table 96: `NVMeControllerProperties.NVMeControllerAttributes`.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsReservations</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Optional NVM Command Support (ONCS): Bit 05 NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 520 : 521: Bit 05
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports namespace reservations.	Indicates whether or not the controller supports reservations.
LongDescription	This property shall indicate whether or not the controller supports namespace reservations to enable two or more hosts to coordinate access to a shared namespace.	Indicates whether or not the controller supports reservations.
Mandatory	Optional Recommended for NVM Drives and more complex devices with NVMe front ends, such as opaque arrays.	Mandatory
Notes		

6.4.3.2.24 NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive` is summarized in Table 97.

Table 97: SupportsTrafficBasedKeepAlive mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeControllerProperties.NVMeControllerAttributes.SupportsTrafficBasedKeepAlive	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 06 - Traffic Based Keep Alive Support (TBKAS) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96 : 99: Bit 06
Description	Boolean Indicates whether or not the controller supports restarting KeepAlive Timer if traffic is processed from an admin command or IO during KeepAlive Timeout interval.	Single bit (bool) Indicates if the controller supports restarting the Keep Alive Timer if an Admin command or an I/O command is processed during the Keep Alive Timeout Interval.
LongDescription	This property shall indicate whether or not the controller supports restarting KeepAlive Timer if traffic is processed from an admin command or IO during KeepAlive Timeout interval.	controller supports restarting the Keep Alive Timer if an Admin command or an I/O command is processed during the Keep Alive Timeout Interval.
Mandatory	Optional Mandatory for Ethernet-Attach Drives; Mandatory for more complex devices with NVMe front ends, such as opaque arrays.	Mandatory

6.4.3.2.25 NVMeControllerProperties.NVMeControllerAttributes.SupportsPredictableLatencyMode The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsPredictableLatencyMode` is summarized in Table 98.

Table 98: `NVMeControllerProperties.NVMeControllerAttributes`.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsPredictableLatencyMode</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 05 - (Predictable Latency Mode) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96:99: Bit 05
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports Predictable Latency Mode.	Indicates if the controller supports Predictable Latency Mode.
LongDescription	This property shall indicate whether or not the controller supports Predictable Latency Mode.	Indicates if the controller supports Predictable Latency Mode
Mandatory	Optional	Mandatory
Notes		

6.4.3.2.26 NVMeControllerProperties.NVMeControllerAttributes.SupportsEnduranceGroups The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsEnduranceGroups` is summarized in Table 99.

Table 99: `NVMeControllerProperties.NVMeControllerAttributes`.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsEnduranceGroups</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 04 (Endurance Groups) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96 : 99: Bit 04
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports Endurance Groups.	Indicates if the controller supports Endurance Groups
LongDescription	This property shall indicate whether or not the controller supports Endurance Groups.	Indicates if the controller supports Endurance Groups
Mandatory	Optional Mandatory when EnduranceGroups/Sets are supported.	Mandatory
Notes	For NVMe SSD Drives: If “Advanced Features for NVMe Drives” feature is advertised, this is required. (This means EnduranceGroups and NVM Sets are supported.)	

6.4.3.2.27 NVMeControllerProperties.NVMeControllerAttributes.SupportsReadRecoveryLevels The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsReadRecoveryLevels` is summarized in Table 100.

Table 100: `NVMeControllerProperties.NVMeControllerAttributes.SupportsReadRecoveryLevels`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsReadRecoveryLevels</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 03 (Read Recovery Levels) NVM Spec Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96: 99: Bit 03
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports Read Recovery Levels.	Indicates whether or not the controller supports Read Recovery Levels.
LongDescription	This property shall indicate whether or not the controller supports Read Recovery Levels.	Indicates whether or not the controller supports Read Recovery Levels.
Mandatory	Optional	Mandatory
Notes		

6.4.3.2.28 NVMeControllerProperties.NVMeControllerAttributes.SupportsNVMSets The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsNVMSets` is summarized in Table 101.

Table 101: `NVMeControllerProperties.NVMeControllerAttributes.SupportsNVMSets`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsNVMSets</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 02 (NVM Sets) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96 : 99: Bit 02
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports NVM Sets.	Indicates whether or not the controller supports NVM Sets.
LongDescription	This property shall indicate whether or not the controller supports NVM Sets.	This property shall indicate whether or not the controller supports NVM Sets.
Mandatory	Optional Mandatory when EnduranceGroups/Sets are supported.	Mandatory
Notes	For NVMe SSD Drives: If “Advanced Features for NVMe Drives” feature is advertised, this is required. (This means EnduranceGroups and NVM Sets are supported.)	

6.4.3.2.29 NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState

The mapping for `NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState` is summarized in Table 102.

Table 102: SupportsExceedingPowerOfNonOperationalState

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeControllerAttributes.SupportsExceedingPowerOfNonOperationalState</code>	NVM Spec Property / Field: Identify Command / Identify Controller Datastructure (CNS 01h) / Controller Attributes (CTRATT): Bit 01 (Non-Operational Power State Permissive Mode) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96 : 99: Bit 01
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports exceeding Power of NonOperational State in order to execute controller initiated background operations in a non-operational power state.	Indicates whether or not the controller supports host control of whether the controller may temporarily exceed the power of a non-operational power state for the purpose of executing controller initiated background operations in a non-operational power state (i.e., Non-Operational Power State Permissive Mode supported)
LongDescription	This property shall indicate whether or not the controller supports exceeding Power of NonOperational State in order to execute controller initiated background operations in a non-operational power state.	The controller supports host control of whether the controller may temporarily exceed the power of a non-operational power state for the purpose of executing controller initiated background operations in a non-operational power state (i.e., Non-Operational Power State Permissive Mode supported)
Mandatory	Optional	Mandatory
Notes		

6.4.3.2.30 NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId The mapping for `NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId` is summarized in Table 103.

Table 103: Supports128BitHostId

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeControllerProperties.NVMeControllerAttributes.Supports128BitHostId	NVM Spec Property / Field: Identify Command / Identify Controller (CNS 01h) / Controller Attributes (CTRATT): Bit 00 (Host Identifier Support) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 96 : 99: Bit 00
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the controller supports a 128-bit Host Identifier.	Indicates whether or not the controller supports a 128-bit Host Identifier
LongDescription	This property shall indicate whether or not the controller supports a 128-bit Host Identifier.	Indicates whether or not the controller supports a 128-bit Host Identifier
Mandatory	Mandatory	Mandatory
Notes		

6.4.3.2.31 NVMeControllerProperties.MaxQueueSize The mapping for `NVMeControllerProperties.MaxQueueSize` is summarized in Table 104.

Table 104: `NVMeControllerProperties.MaxQueueSize` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.MaxQueueSize</code>	NVM Spec Property / Field: <code>JPropertyGet / ControllerProperties / ControllerCapabilities</code> offset 0h (CAP) / Maximum Queues Entries Supported (MQES) NVM Spec: Section:Figure NVMe 2.0; Section 3.1.3.1 Controller Capabilities: Figure 36: Bits 00:15
Type	Int64	16 bit value
Description	Indicates the maximum individual queue size that an NVMe IO Controller supports.	This field indicates the maximum individual queue size that the controller supports.
LongDescription	This property shall contain the maximum individual queue entry size supported per queue. This is a zero-based value, where the minimum value is one, indicating two entries. For PCIe, this applies to both submission and completion queues. For NVMe-oF, this applies to only submission queues.	This field indicates the maximum individual queue size that the controller supports. For NVMe over PCIe implementations, this value applies to the I/O Submission Queues and I/O Completion Queues that the host creates. For NVMe over Fabrics implementations, this value applies to only the I/O Submission Queues that the host creates. This is a 0's based value. The minimum value is 1h, indicating two entries.
Mandatory	Optional Mandatory for more complex devices with NVMe front ends, such as opaque arrays.	Required
Notes		

6.4.3.2.32 NVMeControllerProperties.ANCharacteristics The mapping for `NVMeControllerProperties.ANCharacteristics` is summarized in Table 105.

Table 105: `NNVMeControllerProperties.ANCharacteristics` mapping

	RRedfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.ANCharacteristics</code>	NVM Spec Property / Field: Asymmetric Namespace Access Log (Log ID 0ch) NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.13: Figures 220 and 221
Type	Collection(StorageController.v1_0_0.ANCharacteristics)	Log Page
Description	This property contains the combination of ANA type and volume information.	This log consists of a header describing the log and descriptors containing the asymmetric namespace access information for ANA Groups
LongDescription	This property shall contain the combination of ANA type and volume information.	This log consists of a header describing the log and descriptors containing the asymmetric namespace access information for ANA Groups that contain namespaces that are attached to the controller processing the command. ANA Group Descriptors shall be returned in ascending ANA Group Identifier order
Mandatory	Optional	Mandatory if ANA Reporting is supported
Notes		

6.4.3.2.33 NVMeControllerProperties.ANCharacteristics.AccessState The mapping for `NVMeControllerProperties.ANCharacteristics.AccessState` is summarized in Table 106.

Table 106: `NNVMeControllerProperties.ANCharacteristics.AccessState` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.ANCharacteristics.AccessState</code>	NVM Spec Property / Field:** Asymmetric Namespace Access Log (Log ID 0ch) / ANA Group Descriptors 0 - n-1: Byte 16 Asymmetric Namespace Access State (Bits 00 - 03) NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.13: Figures 220 and 221
Type	<code>StorageController.v1_0_0.ANAAccessState</code>	Reported ANA Access state
Description	Reported ANA Access state.	This field indicates the Asymmetric Namespace Access state for all namespaces in this ANA Group when accessed through this controller.
LongDescription	This property shall contain the reported ANA Access State.	Within the Asymmetric Namespace Access Log (Log ID 0ch), for each ANA Group Descriptor the Asymmetric Namespace Access state for all namespaces in this ANA Group when accessed through this controller.
Mandatory	Optional	Mandatory if ANA Reporting is supported
Notes	Available values: Optimized / NonOptimized / Inaccessible / PersistentLoss	Available values: 01h ANA Optimized state02h ANA Non-Optimized state03h ANA Inaccessible state04h ANA Persistent Loss state0Fh ANA Change state

6.4.3.2.34 NVMeControllerProperties.ANCharacteristics.Volume The mapping for `NVMeControllerProperties.ANCharacteristics.` is summarized in Table 107.

Table 107: `NNVMeControllerProperties.ANCharacteristics.Volume` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.ANCharacteristics.Volume</code>	NVM Spec Property / Field:** Asymmetric Namespace Access Log (Log ID 0ch) / ANA Group Descriptors / Namespace Identifier 0 - n-1 NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.13: Figures 220, 221, and 222
Type	<code>Volume.Volume</code>	ANA Group Descriptor
Description	The specified volume.	The Namespace Identifier
LongDescription	This property shall contain a link to the specified volume.	The Namespace Identifier
Mandatory	Optional	Mandatory if ANA Reporting is supported
Notes	This field contains the pointer to the namespace for which the access state applies.	The namespace id should be linked to the corresponding namespace (volume) object with that namespace id. If this set of fields contains multiple namespaces (e.g., a group of namespaces), a unique entry in the <code>ANCharacteristics</code> array should be created for each namespace.

6.4.3.2.35 NVMeControllerProperties.NVMeSMARTCriticalWarnings.PRMUnreliable The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.PRMUnreliable` is summarized in Table 108.

Table 108: `NNVMeControllerProperties.NVMeSMARTCriticalWarnings.PRMUnreliable` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeSMARTCriticalWarnings.PRMUnreliable</code>	NVM Spec Property / Field: Property Get / Persistent Memory Region Status (PMRSTS) Offset E08h: Bits 09 : 11 - Health Status (HSTS) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.24: Figure 65: Bits 09 : 11 (Health Status / HSTS), value 011b (Unreliable)
Type	Boolean	Single bit (bool)
Description	The Persistent Memory Region has become unreliable.	Persistent Memory Region has become read-only or unreliable
LongDescription	This property shall indicate that the Persistent Memory Region has become unreliable. PCI Express memory reads may return invalid data or generate poisoned PCI Express TLP(s). Persistent Memory Region memory writes may not update memory or may update memory with undefined data. The Persistent Memory Region may also have become non-persistent.	This field shall indicate that the Persistent Memory Region has become read-only or unreliable
Mandatory	Optional Recommended for NVM Drives; required for more complex devices with NVMe front ends, such as opaque arrays.	Optional
Notes		

6.4.3.2.36 NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed` is summarized in Table 109.

Table 109: `NVMeControllerProperties.NVMeSMARTCriticalWarnings`.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeSMARTCriticalWarnings.PowerBackupFailed</code>	NVM Spec Property / Field: SMART / Health Information Log Page (Log ID 02h) / Critical Warning / Volatile memory backup device has failed NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3: Figure 207: Bytes 00, Bits 04
Type	Boolean	Single bit (bool)
Description	Indicates that the volatile memory backup device has failed.	Indicates that the volatile memory backup device has failed.
LongDescription	This property shall indicate that the volatile memory backup device has failed.	This property shall indicate that the volatile memory backup device has failed.
Mandatory	Optional Recommended for NVM Drives. Required for more complex devices with NVMe front ends, such as opaque arrays.	Optional
Notes		

6.4.3.2.37 NVMeControllerProperties.NVMeSMARTCriticalWarnings.MediaInReadOnly The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.MediaInReadOnly` is summarized in Table 110.

Table 110: `NVMeControllerProperties.NVMeSMARTCriticalWarnings.MediaInReadOnly` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeControllerProperties.NVMeSMARTCriticalWarnings.MediaInReadOnly</code>	NVM Spec Property / Field: SMART / Health Information Log Page (Log ID 02h) / Critical Warning / All of the media has been placed in read only mode NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3: Figure 207: Bytes 00, Bits 03
Type	Boolean	Single bit (bool)
Description	Indicates the media has been placed in read only mode.	Indicates all of the media has been placed in read only mode
LongDescription	This property shall indicate the media has been placed in read only mode. This is not set when the read-only condition on the media is a result of a change in the write protection state of a namespace.	This property indicates all of the media has been placed in read only mode. The controller shall not set this bit to '1' if the read-only condition on the media is a result of a change in the write protection state of a namespace
Mandatory	Mandatory	Optional
Notes		

6.4.3.2.38 NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSystemDegraded The mapping for `NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemDegraded` is summarized in Table 111.

Table 111: OverallSystemDegraded mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeControllerProperties.NVMeSMARTCriticalWarnings.OverallSubsystemDegraded	NVM Spec Property / Field: SMART / Health Information Log Page (Log ID 02h) / Critical Warning / NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability. NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3: Figure 207: Bytes 00, Bits 02
Type	Boolean	Single bit (bool)
Description	Indicates that the NVM subsystem reliability has been compromised.	Indicates that NVM subsystem reliability has been degraded.
LongDescription	This property shall indicate that the NVM subsystem reliability has been compromised.	NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability.
Mandatory	Mandatory	Optional
Notes		

6.4.3.2.39 NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut The mapping for NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut is summarized in Table 112.

Table 112: NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeControllerProperties.NVMeSMARTCriticalWarnings.SpareCapacityWornOut	NVM Spec Property / Field: SMART / Health Information Log Page (Log ID 02h) / Critical Warning / available spare capacity has fallen below the threshold. NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3: Figure 207: Bytes 00, Bits 00
Type	Boolean	Single bit (bool)
Description	Indicates that the available spare capacity has fallen below the threshold.	Indicates that the available spare capacity has fallen below the threshold.
LongDescription	This property shall indicate that the available spare capacity has fallen below the threshold.	This property shall indicate that the available spare capacity has fallen below the threshold.
Mandatory	Optional Mandatory for NVMe drives, as well as for more complex devices with NVMe front ends, such as opaque arrays.	Optional
Notes		

6.4.3.2.40 PCIeInterface.PCIeType The mapping for `PCIeInterface.PCIeType` is summarized in Table 113.

Table 113: `PCIeInterface.PCIeType` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PCIeInterface.PCIeType</code>	N/A
Type	enum (<code>PCIeDevice.PCIeType</code>)	N/A
Description	The version of the PCIe specification in use by this device.	N/A
LongDescription	This property shall contain the negotiated PCIe interface version in use by this device.	N/A
Mandatory	Optional Mandatory for PCIe attach NVMe Drives; Do Not Implement for ethernet-attach drives. Optional for more complex devices with NVMe front ends, such as opaque arrays.	
Notes	Possible values: Gen1/Gen2/Gen3/Gen4/Gen5	The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.4.3.2.41 PCIeInterface.MaxPCIeType The mapping for `PCIeInterface.MaxPCIeType` is summarized in Table 114.

Table 114: `PCIeInterface.MaxPCIeType` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PCIeInterface.MaxPCIeType</code>	N/A
Type	enum (<code>PCIeDevice.PCIeType</code>)	N/A
Description	The highest version of the PCIe specification supported by this device.	N/A
LongDescription	This property shall contain the maximum PCIe specification that this device supports.	N/A
Mandatory	Optional Mandatory for PCIe attach NVMe Drives; do not implement for ethernet-attach drives.	
Notes	Possible values: Gen1/Gen2/Gen3/Gen4/Gen5	The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.4.3.2.42 PCIeInterface.LanesInUse The mapping for `PCIeInterface.LanesInUse` is summarized in Table 115.

Table 115: `PCIeInterface.LanesInUse` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PCIeInterface.LanesInUse</code>	N/A
Type	int64	N/A
Description	The number of PCIe lanes in use by this device.	N/A
LongDescription	This property shall contain the number of PCIe lanes in use by this device, which shall be equal to or less than the <code>MaxLanes</code> property value.	N/A
Mandatory	Optional Mandatory for PCIe attach NVMe Drives; do not implement for ethernet-attach drives.	
Notes		The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.4.3.2.43 PCIeInterface.MaxLanes The mapping for `PCIeInterface.MaxLanes` is summarized in Table 116.

Table 116: `PCIeInterface.MaxLanes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PCIeInterface.MaxLanes</code>	N/A
Type	int64	N/A
Description	The number of PCIe lanes supported by this device.	N/A
LongDescription	This property shall contain the maximum number of PCIe lanes supported by this device.	N/A
Mandatory	Optional Mandatory for PCIe attach NVMe Drives; do not implement for ethernet-attach drives.	
Notes		The functionality comes from the underlying implementation and does not originate in the NVMe specs

6.4.3.2.44 Ports The mapping for `Ports` is summarized in Table 117.

Table 117: Ports mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Ports	N/A
Type	PortCollection.PortCollection	N/A
Description	The link to the collection of ports that exist on the storage controller.	N/A
LongDescription	This property shall contain a link to a resource collection of type PortCollection.	N/A
Mandatory	Do Not Implement for NVMe drives Optional for more complex devices with NVMe front ends, such as opaque arrays.	
Notes		

6.4.3.2.45 SKU The mapping for `SKU` is summarized in Table 118.

Table 118: SKU mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SKU	N/A
Type	Edm.String	N/A
Description	The SKU for this storage controller.	N/A
LongDescription	This property shall contain the stock-keeping unit number for this storage storage controller.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.3.2.46 SpeedGbps The mapping for `SpeedGbps` is summarized in Table 119.

Table 119: SpeedGbps mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SpeedGbps	N/A
Type	Decimal	N/A
Description	The maximum speed of the storage controller's device interface.	N/A
LongDescription	This property shall represent the maximum supported speed of the storage bus interface, in Gbit/s. The specified interface connects the controller to the storage devices, not the controller to a host. For example, SAS bus not PCIe host bus.	N/A
Mandatory	Do Not Implement	
Notes		

6.4.3.2.47 Status.State The mapping for `Status.State` is summarized in Table 120.

Table 120: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	NVM Spec Property / Field: Property Get / CSTS – Controller Status (Offset 1ch) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.6: Figure 47
Type	Resource.State (enum)	32-bit value
Description	The known state of the resource, such as, enabled.	
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	
Mandatory	Mandatory	Mandatory
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySpare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified	Ready (CSTS.RDY) maps to Enabled, Shutdown (CSTS.SHST) value will tell you if shutdown is in progress or complete (StandbyOffline), Processing Paused (CSTS.PP) maps to Deferring. If both Ready and Shutdown are indicated, then the system should indicate StandbyOffline. If both Ready and Processing Paused are indicated, then the system should indicate Deferring.

6.4.3.2.48 Status.Health

The mapping for `Status.Health` is summarized in Table 121.

Table 121: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	NVM Spec Property / Field: Property Get command / Controller Status (CSTS): - Offset 1Ch NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.6: Figure 47 NVM Spec Property / Field: Get Log Page command / SMART/Health Information Log (Log Identifier 02h) NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3: Figure 207
Type	Resource.Health	32-bit value
Description	The health state of this resource in the absence of its dependent resources.	This field indicates critical warnings for the state of the controller from both the SMART & general health information log and the Controller Status Property
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	This field indicates critical warnings for the state of the controller from the SMART and general health information . The information provided is over the life of the controller and is retained across power cycles.
Mandatory	Mandatory	Optional

	Redfish/Swordfish	NVMe / NVMe-oF
Notes	Possible Values: OK / Warning / Critical	<p>This comes from CSTS Controller Status. Redfish/Swordfish “OK” corresponds to NVMe/NVMe-oF ‘Ready’ (RDY) bit set to ‘1’ and no warnings in the CSTS or SMART/Health information log page. Redfish/Swordfish “Warning” corresponds to “Volatile Memory Backup Device has failed”, “NVM subsystem reliability has been degraded”, “Temperature greater than or equal to an over temperature threshold”, “temperature less than or equal to an under temperature threshold”, or “available spare capacity has fallen below the threshold”. Redfish/Swordfish “Critical” corresponds to NVMe/NVMe-oF ‘Controller Fatal Status’ (CSTS.CFS) bit set to ‘1’ which indicates that a serious error condition has occurred.</p>

6.4.3.2.49 SupportedControllerProtocols The mapping for `SupportedControllerProtocols` is summarized in Table 122.

Table 122: `SupportedControllerProtocols` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>SupportedControllerProtocols</code>	N/A
Type	<code>Collection(Protocol.Protocol)</code>	N/A
Description	The supported set of protocols for communicating to this storage controller.	N/A
LongDescription	This property shall contain the supported set of protocols for communicating to this storage controller.	N/A
Mandatory	Optional	
Notes	This is an array of protocols supported by the <code>StorageController</code> . This can be set to values including, but not limited to, PCIe, RDMA, NVMe-oF, RoCE, RoCEv2, and InfiniBand.	

6.4.3.2.50 SupportedDeviceProtocols The mapping for SupportedDeviceProtocols is summarized in Table 123.

Table 123: SupportedDeviceProtocols mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SupportedDeviceProtocols	N/A
Type	Collection(Protocol.Protocol)	N/A
Description	The protocols that the storage controller can use to communicate with attached devices.	N/A
LongDescription	This property shall contain the set of protocols this storage controller can use to communicate with attached devices.	N/A
Mandatory	Do Not Implement	
Notes		

6.5 Namespace

The Redfish/Swordfish Volume schema is used to represent an NVM Namespace.

6.5.1 Mockup

```
{
  "@odata.type": "#Volume.v1_9_0.Volume",
  "Id": "1",
  "Name": "Namespace 1",
  "Description": "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.",
  "DisplayName": "Not set",

  "Status": {
    "State": "Enabled",
    "Health": "OK"
  },
  "Identifiers": [{
    "DurableNameFormat": "NQN",
    "DurableName": "nqn.2014-08.org.nvmexpress:uuid:6c5fe566-10e6-4fb6-aad4-8b4159029384"
  }],
  "RemainingCapacityPercent": 100,
  "BlockSizeBytes": 4096,
  "Capacity": {
    "Data": {
      "ConsumedBytes": 0,
      "AllocatedBytes": 10737418240
      "ProvisionedBytes": 10737418240
    }
  },
  "RAIDType": "None",
  "NVMeNamespaceProperties": {
    "IsShareable": false,
    "NamespaceId": "0x22F",
    "NamespaceFeatures": {
      "SupportsThinProvisioning": false,
      "SupportsAtomicTransactionSize": false,
      "SupportsDeallocatedOrUnwrittenLBEError": false,
      "SupportsNGUIDReuse": false,
      "SupportsIOPerformanceHints": false
    }
  },
  "LBAFormat": {
    "LBAFormatType": "LBAFormat0",
    "RelativePerformance": "Best",
    "LBADataSizeBytes": 4096,
  }
}
```

```
    "LBAMetadataSizeBytes": 0
  },
  "MetadataTransferredAtEndOfDataLBA": false,
  "NVMeVersion": "1.4"
},
"Links": {
  "Drives": [{
    "@odata.id": "/redfish/v1/Chassis/SimplestNVMeSSD/Drives/SimplestNVMeSSD"
  }],
  "Controllers": [{
    "@odata.id": "/redfish/v1/Systems/Sys-1/Storage/SimplestNVMeSSD/Controllers/NVMeIOController"
  }]
},
"@odata.id": "/redfish/v1/Systems/Sys-1/Storage/SimplestNVMeSSD/Volumes/SimpleNamespace",
"@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."
}
```

6.5.2 Property Mapping

6.5.2.1 BlockSizeBytes

The mapping for `BlockSizeBytes` is summarized in Table 124.

Table 124: `BlockSizeBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>BlockSizeBytes</code>	<p>NVM Spec Property / Field: Formatted LBA Size (FLBAS) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved NVM Spec:</p> <p>Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 26 - Formatted LBA Size (FLBAS) and uses the Number LBA Formats (NBLAF) field defined in Section 4.1.5.1: Figure 97: Bytes 25 (Number of LBA Formats) NVMe Express NVMe Zoned Namespace Command Set Specification 1.1b: Section A.5: Figure 53 (Size and Capacity Fields): Bytes 26.</p>
Type	Int64	Int 64
Description	The size, in bytes, of the smallest addressable unit, or block.	The LBA data size & metadata size combination that the namespace has been formatted with.
LongDescription	This property shall contain size of the smallest addressable unit of the associated drive or device.	This field indicates the LBA data size & metadata size combination that the namespace has been formatted with.
Mandatory	Mandatory	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

Redfish/Swordfish	NVMe / NVMe-oF
Notes	<p>Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. For NVM Express Command Sets that Support Blocks: Bits 6:5 indicate the most significant 2 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace. If the NLBAF field is less than or equal to 16, then the host should ignore these bits. Bit 4 indicates whether or not the metadata is transferred at the end of the data LBA, creating an extended data LBA. (note: Bit 4 is not applicable when there is no metadata). Bits 3:0 indicate the least significant 4 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace.</p>

6.5.2.2 Capacity.Data.ConsumedBytes The mapping for `Capacity.Data.ConsumedBytes` is summarized in Table 125.

Table 125: Capacity.Data.ConsumedBytes mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Capacity.Data.ConsumedBytes	<p>NVM Spec Property / Field: Namespace Utilization (NUSE) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved.</p> <p>NVM Spec:</p> <p>Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 16 - 23 - Namespace Utilization (NUSE). NVMe Express Key Value Command Set Specification 1.0b: Section 4.1.5.1 - I/O Command Set specific Identify Namespace data structure (CNS 05h, CSI 01h): Figure 36: Bytes 16 -23 (Namespace Utilization)</p>
Type	Int64	Int 64
Description	The number of bytes consumed in this data store for this data type.	The current number of logical blocks allocated in the namespace.
LongDescription	The value shall be the number of logical bytes currently consumed in this data store for this data type.	This field indicates the current number of logical blocks allocated in the namespace. This field is smaller than or equal to the Namespace Capacity. The number of logical blocks is based on the formatted LBA size. For KV Command set: This field indicates the total size of the namespace in bytes. This is the space to store KV keys and KV values. This field is undefined prior to the namespace being formatted.
Mandatory	Mandatory	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

	Redfish/Swordfish	NVMe / NVMe-oF
Notes	Reporting capacity in bytes is the Redfish and Swordfish standard mechanism. Clients expect the capacity information to be reported consistently for these devices, so the calculation here is to convert the NVMe properties (in blocks) to bytes.	Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved

6.5.2.3 Capacity.Data.ProvisionedBytes The mapping for `Capacity.Data.ProvisionedBytes` is summarized in Table 126.

Table 126: `Capacity.Data.ProvisionedBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Capacity.Data.ProvisionedBytes</code>	NVM Spec Property / Field: Namespace Size (NSZE) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 00 - 07 - Namespace Size (NSZE). NVM Express NVM Zoned Namespace Command Set Specification 1.1b: Section A.5: Figure 53: Bytes 0:7 NVM Express Key Value Command Set Specification 1.0b: Section 4.1.5.1: Figure 36: Bytes 00:07
Type	Int64	Int 64
Description	The maximum number of bytes that can be allocated in this data store for this data type.	The total size of the NVM allocated to this namespace.

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	The value shall be the maximum number of bytes that can be allocated in this data store for this data type.	This field indicates the total size of the namespace in logical blocks. A namespace of size n consists of LBA 0 through (n - 1). The number of logical blocks is based on the formatted LBA size. For ZNS: (NSZE-1) indicates the highest possible LBA in the zoned namespace. For KV: This is the space to store KV keys and KV values. This field is undefined prior to the namespace being formatted.
Mandatory	Optional	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.
Notes	This property is required when issuing a create namespace command. It is also required for “change namespace” when modifying the size of the namespace.	Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don’t define this field, it is considered reserved

6.5.2.4 Capacity.Data.AllocatedBytes The mapping for `Capacity.Data.AllocatedBytes` is summarized in Table 127.

Table 127: Capacity.Data.AllocatedBytes mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Capacity.Data.AllocatedBytes	<p>NVM Spec Property / Field: Namespace Capacity (NCAP) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved.</p> <p>NVM Spec: Section: Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 08 - 15 - Namespace Capacity (NCAP). NVMe Express NVM Zoned Namespace Command Set Specification 1.1b: Section A.5: Figure 53 (Size and Capacity Fields): Bytes 08 - 15 (Namespace Capacity).</p>
Type	Int64	Int 64
Description	The number of bytes currently allocated by the storage system in this data store for this data type.	The maximum number of logical blocks that may be allocated in the namespace at any point in time.
Long Description	The value shall be the number of bytes currently allocated by the storage system in this data store for this data type.	For NVM Command Set: The maximum number of logical blocks that may be allocated in the namespace at any point in time. For Zoned Namespace Command Set: The maximum number of allocatable logical blocks in the zoned namespace.
Mandatory	Optional	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

	Redfish/Swordfish	NVMe / NVMe-oF
Notes	Reporting capacity in bytes is the Redfish and Swordfish standard mechanism. Not required when creating a namespace; service implementations will provide a suitable value. For non-thin-provisioned volumes, this will typically be equal to <code>ProvisionedBytes</code> .	Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved

6.5.2.5 Capacity.Metadata The mapping for `Capacity.Metadata` is summarized in Table 128.

Table 128: `Capacity.Metadata.AllocatedBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Capacity.Metadata</code>	N/A
Type	<code>Int64</code>	N/A
Description	The number of bytes currently allocated by the storage system in this data store for this data type.	N/A
LongDescription	The value shall be the number of bytes currently allocated by the storage system in this data store for this data type.	N/A
Mandatory	Optional	
Notes	Not required for NVMe Drives; optional to implement for more complex devices.	Do not return metadata information for NVMe devices. This is included in the overall reported capacity information.

6.5.2.6 CapacitySources The mapping for `CapacitySources` is summarized in Table 6.5.2.6.

```
+--+-----+-----+ || Redfish/Swordfish | NVMe / NVMe-oF | +:==+:=====+:=====+
```

: `CapacitySources` mapping

Property | `CapacitySources` | See notes |

Type | `Collection(Capacity.CapacitySource)` | See notes |

Description | An array of space allocations to this volume. | See notes |

LongDescription | Fully or partially consumed storage from a source resource. Each entry provides capacity allocation information from a named source resource. | See notes. |

Mandatory | No | No |

Notes | Contains the information about the providing capacity (e.g, EnduranceGroup) for this namespace. Each entry in the collection is mapped such that the CapacitySource instance contains the overall capacity (in bytes) and the types and pointers to the underlying capacity sources. | For each entry in the CapacitySource collection, map the information from the NVMe specification: **NVM Spec Property / Field:** Supported Capacity Configuration List (Log Identifier 11h)**NVM Spec: Section:Figure** NVMe Base Specification 2.0: Section 5.16.1.17: Figures 250 and 251. |

6.5.2.7 Description The mapping for `Description` is summarized in Table 129.

Table 129: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Description is a read-only field.	Return the common description: “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.”

6.5.2.8 DisplayName The mapping for `DisplayName` is summarized in Table 130.

Table 130: DisplayName mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	DisplayName	N/A
Type	String	N/A

	Redfish/Swordfish	NVMe / NVMe-oF
Description	A user-configurable string to name the volume.	N/A
LongDescription	This property shall contain a user-configurable string to name the volume.	N/A
Mandatory	Optional Recommended for NVMe Drives, as well as for more complex devices with NVMe front ends, such as opaque arrays.	
Notes	This contains an (end) user settable “friendly” name for the namespace. This may map to a property that shows up in the proprietary software, as long as it is configurable by the end-user, and is intended to contain a displayable string.	

6.5.2.9 Identifiers The mapping for `Identifiers` is summarized in Table 131.

Table 131: Identifiers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers	NVM Spec Property / Field: Identify / Namespace Identification Descriptor list (CNS 03h) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.3, Figure 277
Type	Collection(Resource.Identifier)	A variable length Namespace Identification Descriptor structures
Description	The Durable names for the storage controller.	A list of Namespace Identification Descriptor structures containing Namespace Type (i.e., IEEE Extended Unique Identifier, Namespace Globally Unique Identifier, Namespace UUID, Command Set Identifier), Namespace Identifier Length (NIDL), and Namespace ID (NID).

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This property shall contain a list of all known durable names for the associated storage controller.	A list of any number of variable length Namespace Identification Descriptor structures that fit into the 4,096 byte Identify payload. All remaining bytes after the namespace identification descriptor structures should be cleared to 0h, and the host shall interpret a Namespace Identifier Descriptor Length (NIDL) value of 0h as the end of the list. The host should ignore any Namespace Identification Descriptor with a Namespace Identifier Type not supported by the host.
Mandatory	Optional	Yes
Notes	This is an array of unique identifiers for the NVM Subsystem including Namespace Type and Namespace ID.	

6.5.2.10 Identifiers.DurableName The mapping for `Identifiers.DurableName` is summarized in Table 132.

Table 132: `Identifiers.DurableName` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Identifiers.DurableName</code>	NVM Spec Property / Field: Identify / Namespace Identification Descriptor list (CNS 03h) / Namespace Identifier (NID) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.3, Figure 277: Bytes 04 : (NIDL + 3)
Type	Variable - see notes	Variable: See notes
Description	The Durable names for the storage controller.	This field contains a value that is globally unique and assigned to the namespace when the namespace is created.

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This property shall contain a list of all known durable names for the Namespace.	This field contains a value that is globally unique and assigned to the namespace when the namespace is created. This field remains fixed throughout the life of the namespace and is preserved across namespace and controller operations (e.g., Controller Level Reset, namespace format, etc.). The type of the value is specified by the Namespace Identifier Type (NIDT) field, and the size is specified by the Namespace Identifier Length (NIDL) field.
Mandatory	Optional	Yes
Notes	This is an array of unique identifiers for the Namespace. Type and length of descriptor are in the corresponding Identifiers.DurableNameFormat property.	The type and length of the descriptor are in the corresponding Identifiers.DurableNameFormat property

6.5.2.11 Identifiers.DurableNameFormat The mapping for `Identifiers.DurableNameFormat` is summarized in Table 133.

Table 133: Identifiers.DurableNameFormat mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers.DurableNameFormat	NVM Spec Property / Field: Identify / Namespace Identification Descriptor list (CNS 03h) / Namespace Identifier Type (NIDT) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.3, Figure 277: Bytes 00
Type	Int64	Int64
Description	The Durable names for the storage controller.	The Namespace Identifier data type
LongDescription	This property shall contain a list of the types for all known durable names for the associated storage controller. The type determines the length of the corresponding Namespace ID	The data type contained in the Namespace Identifier field and the length of that type.
Mandatory	Optional	Yes

	Redfish/Swordfish	NVMe / NVMe-oF
Notes	This is an array of types for the unique identifiers for the NVM Subsystem. Values may be 'EUI64', 'NGUID', or 'UUID'.	Allowed values:1h = an 8-byte IEEE Extended Unique Identifier2h = a 10-byte Namespace Globally Unique Identifier.3h = an 8-byte Namespace UUID.4h = the Command Set Identifier (the command set that operates on this namespace).

6.5.2.12 InitializeMethod The mapping for `InitializeMethod` is summarized in Table 134.

Table 134: InitializeMethod mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	InitializeMethod	N/A
Type	Volume.InitializeMethod (enum)	N/A
Description	Indicates the Initialization Method used for this volume. If InitializeMethod is not specified, the InitializeMethod should be Foreground.	N/A
LongDescription	This property shall indicate the initialization method used for this volume. If InitializeMethod is not specified, the InitializeMethod should be Foreground. This value reflects the most recently used Initialization Method, and may be changed using the Initialize Action.	N/A
Mandatory	Optional Recommended for NVMe Drives, as well as for more complex devices with NVMe front ends, such as opaque arrays.	
Notes	Available values: Fast / Slow	Not in NVMe Specification today. This maps to functionality corresponding to traditional drive capabilities outside the scope of NVMe.

6.5.2.13 Links.Controllers The mapping for `Links.Controllers` is summarized in Table 135.

Table 135: Links.Controllers

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.Controllers	N/A
Type	Collection(StorageController.StorageController)	N/A
Description	An array of the Controllers associated with this volume.	N/A
LongDescription	This parameter shall contain an array of the controllers (of type StorageController) associated with this volume. When the volume is of type NVMe, these may be both the physical and logical controller representations.	N/A
Mandatory	Optional Mandatory for NVMe Drives and well as for more complex devices with an NVMe front end, such as opaque arrays.	
Notes	This array shall contain links to the controllers for this namespace.	These are the Redfish constructs that correspond to the Admin and IO Controllers associated with this namespace.

6.5.2.14 Links.Drives The mapping for `Links.Drives` is summarized in Table 136.

Table 136: Links.Drives mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.Drives	N/A
Type	Collection(Drive.Drive)	N/A
Description	An array of the drives to be used by the volume	N/A
LongDescription	This parameter shall contain an array of the drives to be used by the volume.	N/A

	Redfish/Swordfish	NVMe / NVMe-oF
Mandatory	Optional Mandatory for NVMe Drives. Not recommended for more complex devices with an NVMe front end, such as opaque arrays; for these configurations the mapping is to the underlying storage pool, rather than to drives.	
Notes	This array shall contain links to the drive object for this namespace.	Redfish construct. Not in the NVMe spec.

6.5.2.15 LogicalUnitNumber The mapping for `LogicalUnitNumber` is summarized in Table 137.

Table 137: `LogicalUnitNumber` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>LogicalUnitNumber</code>	N/A
Type	<code>Int64</code>	N/A
Description	Indicates the host-visible <code>LogicalUnitNumber</code> assigned to this Volume.	N/A
LongDescription	This property shall contain host-visible <code>LogicalUnitNumber</code> assigned to this Volume. This property shall only be used when in a single connect configuration and no <code>StorageGroup</code> configuration is used.	N/A
Mandatory	Optional	
Notes	Do not use with NVMe devices. This is represented more correctly with <code>(NVMeNamespaceProperties).NamespaceId</code> .	Do not implement.

6.5.2.16 MaxBlockSizeBytes The mapping for `MaxBlockSizeBytes` is summarized in Table 138.

Table 138: MaxBlockSizeBytes mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	MaxBlockSizeBytes	<p>NVM Spec Property / Field: Formatted LBA Size (FLBAS) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved</p> <p>NVM Spec: Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 26 - Formatted LBA Size (FLBAS) and uses the Number LBA Formats (NBLAF) field defined in Section 4.1.5.1: Figure 97: Bytes 25 (Number of LBA Formats) NVMe Express NVM Zoned Namespace Command Set Specification 1.1b: Section A.5: Figure 53 (Size and Capacity Fields): Bytes 26.</p>
Type	Int64	
Description	The size, in bytes, of the smallest addressable unit, or block.	The LBA data size & metadata size combination that the namespace has been formatted with.
LongDescription	This property shall contain size of the smallest addressable unit of the associated drive or device.	This field indicates the LBA data size & metadata size combination that the namespace has been formatted with.
Mandatory	Optional Recommended to not implement; however, if implemented this should have the same value as BlockSizeBytes.	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

Redfish/Swordfish	NVMe / NVMe-oF
No tesNotes	<p>Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. Follow index in 247: Byte 26 (bits 3:0) to the LBA format structure to get the metadata size and LBA data size combination. These combined values are BlockSizeBytes. For NVM Express Command Sets that Support Blocks: Bits 6:5 indicate the most significant 2 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace. If the NLBAF field is less than or equal to 16, then the host should ignore these bits. Bit 4 indicates whether or not the metadata is transferred at the end of the data LBA, creating an extended data LBA. (note: Bit 4 is not applicable when there is no metadata). Bits 3:0 indicate the least significant 4 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace.</p>

6.5.2.17 Name The mapping for `Name` is summarized in Table 139.

Table 139: Name mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Name	N/A
Type	String	N/A
Description	The name of the resource or array member.	N/A

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Name is a read-only field.	

6.5.2.18 NVMeNamespaceProperties.NamespaceId The mapping for `NVMeNamespaceProperties.NamespaceId` is summarized in Table 140.

Table 140: `NVMeNamespaceProperties.NamespaceId` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeNamespaceProperties.NamespaceId</code>	NVM Spec Property / Field: Identify / Namespace Identification Descriptor list (CNS 03h) / Namespace Identifier (NID) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.3, Figure 277: Bytes 04 : (NIDL + 3)
Type	String	Variable: See notes
Description	The NVMe Namespace Identifier for this namespace.	This field contains a value that is globally unique and assigned to the namespace when the namespace is created.
LongDescription	This property shall contain the NVMe Namespace Identifier for this namespace. This property shall be a hex value. Namespace identifiers are not durable and do not have meaning outside the scope of the NVMe subsystem. NSID 0x0, 0xFFFFFFFF, 0xFFFFFFFFE are special purpose values.	This field contains a value that is globally unique and assigned to the namespace when the namespace is created. This field remains fixed throughout the life of the namespace and is preserved across namespace and controller operations (e.g., Controller Level Reset, namespace format, etc.). The type of the value is specified by the Namespace Identifier Type (NIDT) field, and the size is specified by the Namespace Identifier Length (NIDL) field.
Mandatory	Mandatory	Yes

	Redfish/Swordfish	NVMe / NVMe-oF
Notes		The type and length of the descriptor are in the corresponding identifiers. DurableNameFormat property

6.5.2.19 NVMeNamespaceProperties.IsBootCapable The mapping for `NVMeNamespaceProperties.IsBootCapable` is summarized in Table 141.

Table 141: `NVMeNamespaceProperties.IsBootCapable` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeNamespaceProperties.IsBootCapable</code>	NVM Spec Property / Field: Property Get Command / Controller Capabilities - CAP (Offset 00h) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3.1 (Controller Capabilities), Figure 36: Bits 45 - Boot Partition Support (BPS)
Type	Boolean	Single bit (bool)
Description	This property indicates whether or not the Volume contains a boot image and is capable of booting.	This bit indicates whether the controller supports Boot Partitions
Long Description	This property shall indicate whether or not the Volume contains a boot image and is capable of booting. This property may be settable by an admin or client with visibility into the contents of the volume. This property should only be set to true when <code>VolumeUsage</code> is either not specified, or when <code>VolumeUsage</code> is set to <code>Data</code> or <code>SystemData</code> .	This bit indicates whether the controller supports Boot Partitions. If this bit is set to '1', the controller supports Boot Partitions. If this bit is cleared to '0', the controller does not support Boot Partitions
Mandatory	Optional	No
Notes		

6.5.2.20 NVMeNamespaceProperties.IsShareable The mapping for `NVMeNamespaceProperties.IsShareable` is summarized in Table 142.

Table 142: NVMeNamespaceProperties.IsShareable mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeName spaceProperties.IsShareable	NVM Spec Property / Field: Identify command / I/O Command Set Independent Identify Namespace data structure (CNS 08h) / Namespace Multi-path I/O and Namespace Sharing Capabilities (NMIC) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.8 (I/O Command Set Independent Identify Namespace Data Structure), Figure 280: Byte 01 - Namespace Multi-path I/O and Namespace Sharing Capabilities (NMIC)
Type	Boolean	Single bit (bool)
Description	Indicates the namespace is shareable.	Indicates if the namespace is a private namespace and is able to be attached to only one controller at a time or if the namespace may be attached to two or more controllers in the NVM subsystem concurrently (i.e., may be a shared namespace)
LongDescription	This property shall indicate whether the namespace is shareable.	If set to '1', then the namespace may be attached to two or more controllers in the NVM subsystem concurrently (i.e., may be a shared namespace). Bits 7:1 are reserved. Bit 0, if cleared to '0', indicates the namespace is a private namespace and is able to be attached to only one controller at a time.
Mandatory	Optional	No
Notes		

6.5.2.21 NVMeNamespaceProperties.NamespaceFeatures.SupportsThinProvisioning The mapping for NVMeName-
spaceProperties.NamespaceFeatures.SupportsThinProvisioning is summarized in Table 6.5.2.21.

Table 143: NVMeNamespaceProperties.NamespaceFeatures.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeNamespaceProperties.NamespaceFeatures.SupportsThinProvisioning	NVM Spec Property / Field: Namespace Features (NSFEAT) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section: Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 24 - Namespace Features (NSFEAT), Bit 0 (ThinP).
Type	Boolean	Single bit (bool)
Description	This property indicates whether or not the NVMe Namespace supports thin provisioning.	Indicates that the namespace supports thin provisioning
Long Description	This property shall indicate whether or not the NVMe Namespace supports thin provisioning. Specifically, the namespace capacity reported may be less than the namespace size.	if set to '1' indicates that the namespace supports thin provisioning. If cleared to '0' indicates that thin provisioning is not supported.
Mandatory	Mandatory	Yes
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

SupportsThinProvisioning mapping

6.5.2.22 NVMeNamespaceProperties.NamespaceFeatures.SupportsDeallocatedOrUnwrittenLBEError The mapping for NVMeNamespaceProperties.NamespaceFeatures.SupportsDeallocatedOrUnwrittenLBEError is summarized in Table 6.5.2.22.

Table 144: NVMeNamespaceProperties.NamespaceFeatures.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeNamespaceProperties.NamespaceFeatures.SupportsDeallocatedOrUnwrittenLbError	NVM Spec Property / Field: Namespace Features (NSFEAT) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec Section: Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 24 - Namespace Features (NSFEAT), Bit 2 (DAE).
Type	Boolean	Single bit (bool)
Description	This property indicates that the controller supports deallocated or unwritten logical block error for this namespace.	Indicates if the controller supports the Deallocated or Unwritten Logical Block error for this namespace.
LongDescription	This property shall indicate that the controller supports deallocated or unwritten logical block error for this namespace.	If set to '1' indicates that the controller supports the Deallocated or Unwritten Logical Block error for this namespace. If cleared to '0', then the controller does not support the Deallocated or Unwritten Logical Block error for this namespace.
Mandatory	Mandatory	Yes
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

SupportsDeallocatedOrUnwrittenLbError mapping

6.5.2.23 NVMeNamespaceProperties.NamespaceFeatures.SupportsNGUIDReuse The mapping for `NVMeNamespaceProperties.NamespaceFeatures.SupportsNGUIDReuse` is summarized in Table 145.

Table 145: NVMeNamespaceProperties.NamespaceFeatures.SupportsNGUIDReuse mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeNamespaceProperties.NamespaceFeatures.SupportsNGUIDReuse	NVM Spec Property / Field: Namespace Features (NSFEAT) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 24 - Namespace Features (NSFEAT), Bit 3 (UIDRESUSE). NVMe KV Command Set Specification 1.0b: Section 4.1.5.1: Figure 36: Bytes 24 - Namespace Features (NSFEAT), Bit 3 (UIDRESUSE).
Type	Boolean	Single bit (bool)
Description	This property indicates that the namespace supports the use of an NGUID (namespace globally unique identifier) value.	Indicates if the value in the NGUID field and the value in the EUI64 field for this namespace may be reused by the controller for a new namespace created after this namespace is deleted.
LongDescription	This property shall indicate that the namespace supports the use of an NGUID (namespace globally unique identifier) value.	If set to '1' indicates that the value in the NGUID field for this namespace, if non-zero, is never reused by the controller and that the value in the EUI64 field for this namespace, if non-zero, is never reused by the controller. If cleared to '0', then the NGUID value may be reused and the EUI64 value may be reused by the controller for a new namespace created after this namespace is deleted. This bit shall be cleared to '0' if both NGUID and EUI64 fields are cleared to 0h.
Mandatory	Mandatory	Yes

	Redfish/Swordfish	NVMe / NVMe-oF
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set, and the KV Command Set. Prohibited for all other I/O Command Sets.

6.5.2.24 NVMeNamespaceProperties.NamespaceFeatures.SupportsAtomicTransactionSize The mapping for `NVMeNamespaceProperties.NamespaceFeatures.SupportsAtomicTransactionSize` is summarized in Table 6.5.2.24.

Table 146: `NVMeNamespaceProperties.NamespaceFeatures`.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeNamespaceProperties.NamespaceFeatures.SupportsAtomicTransactionSize</code>	NVM Spec Property / Field: Namespace Features (NSFEAT) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 24 - Namespace Features (NSFEAT), Bit 4 (OPTPERF).
Type	Boolean	Single bit (bool)
Description	Indicates whether or not the NVM fields for Namespace preferred write granularity (NPWG), write alignment (NPWA), deallocate granularity (NPDG), deallocate alignment (NPDA) and optimal write size (NOWS) are defined for this namespace and should be used by the host for I/O optimization.	Indicates support for the fields NPWG, NPWA, NPDG, NPDA, and NOWS for this namespace; and optimal Write Size field in NVM Sets Attributes Entry

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This property shall indicate whether or not the NVM fields for Namespace preferred write granularity (NPWG), write alignment (NPWA), deallocate granularity (NPDG), deallocate alignment (NPDA) and optimal write size (NOWS) are defined for this namespace and should be used by the host for I/O optimization.	If set to '1' indicates that the fields NAWUN, NAWUPF, and NACWU are defined for this namespace and should be used by the host for this namespace instead of the AWUN, AWUPF, and ACWU fields in the Identify Controller data structure. If cleared to '0', then the controller does not support the fields NAWUN, NAWUPF, and NACWU for this namespace. In this case, the host should use the AWUN, AWUPF, and ACWU fields defined in the Identify Controller data structure.
Mandatory	Mandatory	Yes
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

SupportsAtomicTransactionSize mapping

6.5.2.25 NVMeNamespaceProperties.NamespaceFeatures.SupportsIOPerformanceHints

The mapping for `NVMeNamespaceProperties.NamespaceFeatures.SupportsIOPerformanceHints` is summarized in Table 6.5.2.25.

Table 147: NVMeNamespaceProperties.NamespaceFeatures.

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeNamespaceProperties.NamespaceFeatures.SupportsIOPerformanceHints	NVM Spec Property / Field: Namespace Features (NSFEAT) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section: Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 24 - Namespace Features (NSFEAT), Bit 1 (NSABP).
Type	Boolean	Single bit (bool)
Description	Indicates that the Namespace Atomic Write Unit Normal (NAWUN), Namespace Atomic Write Unit Power Fail (NAWUPF), and Namespace Atomic Compare and Write Unit (NACWU) fields are defined for this namespace and should be used by the host for this namespace instead of the controller-level properties AWUN, AWUPF, and ACWU.	indicates whether or not the fields NAWUN, NAWUPF, and NACWU are defined for this namespace and should be used by the host for this namespace instead of the AWUN, AWUPF, and ACWU fields in the Identify Controller data structure.
Long Description	This property shall indicate that the Namespace Atomic Write Unit Normal (NAWUN), Namespace Atomic Write Unit Power Fail (NAWUPF), and Namespace Atomic Compare and Write Unit (NACWU) fields are defined for this namespace and should be used by the host for this namespace instead of the controller-level properties AWUN, AWUPF, and ACWU.	If set to '1' indicates that the fields NAWUN, NAWUPF, and NACWU are defined for this namespace and should be used by the host for this namespace instead of the AWUN, AWUPF, and ACWU fields in the Identify Controller data structure. If cleared to '0', then the controller does not support the fields NAWUN, NAWUPF, and NACWU for this namespace. In this case, the host should use the AWUN, AWUPF, and ACWU fields defined in the Identify Controller data structure
Mandatory	Mandatory	Yes

	Redfish/Swordfish	NVMe / NVMe-oF
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

SupportsIOPerformanceHints mapping

6.5.2.26 NVMeNamespaceProperties.NumberLBAFormats The mapping for `NVMeNamespaceProperties.NumberLBAFormats` is summarized in Table 148.

Table 148: 163: `NVMeNamespaceProperties.NumberLBAFormats` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeNamespace Properties.NumberLBAFormats	NVM Spec Property / Field: Number of LBA Formats (NLBAF) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 25 - Namespace Features (NSFEAT).
Type	Int64	Int64
Description	The number of LBA data size and metadata size combinations supported by this namespace. The value of this property is between 0 and 16.	The number of supported LBA data size and metadata size combinations supported by the namespace.
LongDescription	This property shall contain the number of LBA data size and metadata size combinations supported by this namespace. The value of this property is between 0 and 16. LBA formats with an index set beyond this value will not be supported.	This field defines the number of supported LBA data size and metadata size combinations supported by the namespace. LBA formats shall be allocated in order (starting with 0) and packed sequentially.

	Redfish/Swordfish	NVMe / NVMe-oF
Mandatory	Mandatory	Yes
Notes		This is a 0's based value. The maximum number of LBA formats that may be indicated as supported is: a) 16 if the LBA Format Extension Enable (LBAFEE) field is cleared to 0h in the Host Behavior Support feature (refer to the Host Behavior Support section in the NVMe Base Specification); or b) 64 if the LBAFEE field is set to 1h in the Host Behavior Support feature (refer to the Host Behavior Support section in the NVMe Base Specification). Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

6.5.2.27 NVMeNamespaceProperties.FormattedLBASize The mapping for `NVMeNamespaceProperties.FormattedLBASize` is summarized in Table 149.

Table 149: NVMeNamespaceProperties.FormattedLBASize mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeNamespaceProperties.FormattedLBASize	NVM Spec Property / Field: Formatted LBA Size (FLBAS) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 26 - Formatted LBA Size (FLBAS). NVMe Zoned Namespace Command Set Specification 1.1b: A.5: Figure 53: Byte 26 - Formatted LBA Size (FLBAS).

	Redfish/Swordfish	NVMe / NVMe-oF
Type	Int64	Int64
Description	The LBA data size and metadata size combination that the namespace has been formatted with.	The LBA data size and metadata size combination that the namespace has been formatted with.
LongDescription	This property shall contain the LBA data size and metadata size combination that the namespace has been formatted with. This is a 4-bit data structure.	This field indicates the LBA data size & metadata size combination that the namespace has been formatted with. Bits 3:0 indicates one of the 16 supported LBA Formats indicated in this data structure.
Mandatory	Mandatory	Yes
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

6.5.2.28 NVMeNamespaceProperties.MetadataTransferredAtEndOfDataLBA The mapping for `NVMeNamespaceProperties.MetadataTransferredAtEndOfDataLBA` is summarized in Table 150.

Table 150: `NVMeNamespaceProperties.MetadataTransferredAtEndOfDataLBA` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeNamespaceProperties.MetadataTransferredAtEndOfDataLBA</code>	NVM Spec Property / Field: Metadata Capabilities (MC) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec: Section: Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 27 - Metadata Capabilities (MC).
Type	Boolean	2 Bits

	Redfish/Swordfish	NVMe / NVMe-oF
Description	This property indicates whether or not the metadata is transferred at the end of the LBA creating an extended data LBA.	This property indicates whether or not the metadata is transferred at the end of the data LBA.
LongDescription	This property shall indicate whether or not the metadata is transferred at the end of the LBA creating an extended data LBA.	Bit 1 if set to '1' indicates the namespace supports the metadata being transferred as part of a separate buffer that is specified in the Metadata Pointer. Bit 1 if cleared to '0' indicates that the namespace does not support the metadata being transferred as part of a separate buffer. Bit 0 if set to '1' indicates that the namespace supports the metadata being transferred as part of an extended data LBA. Bit 0 if cleared to '0' indicates that the namespace does not support the metadata being transferred as part of an extended data LBA.
Mandatory	Mandatory	Yes
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

6.5.2.29 NVMeNamespaceProperties.NVMeVersion The mapping for `NVMeNamespaceProperties.NVMeVersion` is summarized in Table 151.

Table 151: `NVMeNamespaceProperties.NVMeVersion` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeNamespaceProperties.NVMeVersion</code>	NVM Spec Property / Field: Property Get Command / Version (VS) - Offset 8h NVM Spec: Section:Figure NVMe Base Specification 2.0: Section 3.1.3.2
Type	String	Int64

	Redfish/Swordfish	NVMe / NVMe-oF
Des cription	The version of the NVMe Base Specification supported.	This property indicates the major, minor, and tertiary version of the NVM Express base specification that the controller implementation supports
LongDes cription		This property indicates the major, minor, and tertiary version of the NVM Express base specification that the controller implementation supports. Valid versions of the specification are: 1.0, 1.1, 1.2, 1.2.1, 1.3, 1.4, and 2.0.
M andatory	Mandatory	Yes
Notes		

6.5.2.30 OptimumIOSizeBytes The mapping for `OptimumIOSizeBytes` is summarized in Table 152.

Table 152: `OptimumIOSizeBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>OptimumIOSizeBytes</code>	NVM Spec Property / Field: Namespace Optimal Write Size (NOWS) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. NVM Spec Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 72 - 73 - Namespace Optimal Write Size (NOWS)
Type	Int64	2 Bytes
Des cription	The size in bytes of this Volume's optimum IO size.	This field indicates the size in logical blocks for optimal write performance for this namespace.

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This property shall contain the optimum IO size to use when performing IO on this volume. For logical disks, this is the stripe size. For physical disks, this describes the physical sector size.	This field indicates the size in logical blocks for optimal write performance for this namespace. This is a 0's based value. If the OPTPERF bit is cleared to '0', then this field is reserved. The size indicated should be less than or equal to Maximum Data Transfer Size (MDTS) that is specified in units of minimum memory page size. The value of this field may change if the namespace is reformatted. The value of this field should be a multiple of Namespace Preferred Write Granularity (NPWG).
Mandatory	Optional Recommended for implementations with Endurance Groups and NVM Sets.	Optional
Notes		Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets. Convert from blocks to bytes. If the namespace is associated with an NVM set, NOWS defined for this namespace shall be set to the Optimal Write Size field setting defined in NVM Set Attributes Entry (refer to the Namespace Identification Descriptor in the NVMe Base Specification) for the NVM Set with which this namespace is associated. If NOWS is not supported, the Optimal Write Size field in NVM Sets Attributes Entry (refer to the Namespace Identification Descriptor in the NVMe Base Specification) for the NVM Set with which this namespace is associated should be used by the host for I/O optimization. Refer to the NVMe NVM Command Set Specification, section 5.8.2 for how this field is utilized to improve performance and endurance

6.5.2.31 ProvisioningPolicy The mapping for `ProvisioningPolicy` is summarized in Table 153.

Table 153: `OptimumIOSizeBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	ProvisioningPolicy	NVM Spec Property / Field: Namespace Features (NSFEAT) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved NVM Spec: Section: Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 24 - Namespace Features (NSFEAT), Bit 0 (ThinP).
Type	enum (DataStorageLoSCapabilities.ProvisioningPolicy)	Single Bit (BOOL)
Description	This property specifies the volume's storage allocation, or provisioning policy.	Indicates that the namespace supports thin provisioning
LongDescription	This property shall specify the volume's supported storage allocation policy.	if set to '1' indicates that the namespace supports thin provisioning. If cleared to '0' indicates that thin provisioning is not supported
Mandatory	Optional Recommended for implementations that support thin provisioning.	Mandatory
Notes	possible values: Fixed / Thin	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

6.5.2.32 Status.State The mapping for `Status.State` is summarized in Table 154.

Table 154: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	NVM Spec Property / Field: Identify Command / I/O Command Set Independent Identify Namespace data structure (CNS 08h) / Namespace Status(NSTAT) NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.8: Figure 280 - Byte14: Namespace Status (NSTAT): - Bit 0: Namespace Ready (NRDY)
Type	Resource.State (enum)	Boolean
Description	The known state of the resource, such as, enabled.	Indicates if the Namespace is 'ready'
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	This field indicates the status of the namespace with the specified NSID. A value of '1' indicates that the namespace is ready. A value of '0' indicates that the namespace is not ready.
Mandatory	Mandatory	Yes
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified	Namespace Ready (NRDY value = 1) maps to 'Enabled'. Namespace Not Ready (NRDY value = 0) maps to 'Disabled'.

6.5.2.33 Status.Health The mapping for `Status.Health` is summarized in Table 155.

Table 155: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	Property Get / Controller Status (Offset 18h) / Controller Fatal Status (CSTS.CFS) - Bit 01 NVM Spec: Section:Figure NVMe 2.0: 3.1.3.6: Figure 47
Type	Resource.Health	Bit
Description	The health state of this resource in the absence of its dependent resources.	Indicates if the controller is able to communicate with host software via completion queue entries in the Admin Completion Queue or I/O Completion Queues.
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	If the controller has a serious error condition and is unable to communicate with host software via completion queue entries in the Admin Completion Queue or I/O Completion Queues, then the controller may set the Controller Fatal Status (CSTS.CFS) bit to '1' (refer to section 3.1.6). This indicates to host software that a serious error condition has occurred. When this condition occurs, host software should attempt to reset and then re-initialize the controller. The Controller Fatal Status condition is not indicated with an interrupt. If host software experiences timeout conditions and/or repeated errors, then host software should consult the Controller Fatal Status (CSTS.CFS) bit to determine if a more serious error has occurred.
Mandatory	Mandatory	Yes
Notes	Possible Values: OK / Warning / Critical	Reference Figure 222 (NVM Subsystem Hardware Error Event Codes) bit 09h of the NVMe Base Specification.

6.5.2.34 Status.HealthRollup The mapping for `Status.HealthRollup` is summarized in Table 156.

Table 156: Status.HealthRollup mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.HealthRollup	N/A
Type	Resource.Health	N/A
Description	The overall health state from the view of this resource.	N/A
LongDescription	This property shall represent the health state of the resource and its dependent resources. The values shall conform to those defined in the Redfish Specification.	N/A
Mandatory	Optional	
Notes	Do not use for Namespace. There are no dependent resources.	Do not implement.

6.5.2.35 StorageGroups

The mapping for `StorageGroups` is summarized in Table 157.

Table 157: StorageGroups mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	StorageGroups	N/A
Type	StorageGroupCollection.StorageGroupCollection	N/A
Description	An array of references to Storage Groups that includes this volume.	N/A
LongDescription	The value of this property shall contain references to all storage groups that include this volume.	N/A
Mandatory	Do Not Implement	
Notes		

6.5.2.36 WriteCachePolicy

The mapping for WriteCachePolicy is summarized in Table 158.

Table 158: WriteCachePolicy mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	WriteCachePolicy	N/A
Type	StorageGroupCollection.StorageGroupCollection	N/A
Description	Indicates the write cache policy setting for the Volume	N/A
LongDescription	This property shall contain a boolean indicator of the write cache policy for the Volume.	N/A
Mandatory	Recommended	
Notes	Possible Values: WriteThrough, ProtectedWriteBack, UnprotectedWriteBack, Off. Set to “Off” when write cache disabled; set to other values when enabled.	

6.6 Endurance Group

The Redfish/Swordfish Storage Pool schema is used to represent an NVM Endurance Group.

6.6.1 Mockup

The following mockup shows a sample representation of the Storage Pool schema used to represent an NVM Endurance Group.

```
{
  "@odata.type": "#StoragePool.v1_7_1.StoragePool",
  "Id": "1",
  "Name": "Endurance Group 1",
  "Description": "Single Endurance Group",
  "NVMeProperties": {
    "NVMePoolType": "EnduranceGroup"
  },
  "NVMeEnduranceGroupProperties": {
    "EndGrpLifetime": {
      "PercentUsed": 0,
      "EnduranceEstimate": 0,
      "DataUnitsRead": 0,
      "DataUnitsWritten": 0,
      "MediaUnitsWritten": 0,
      "HostReadCommandCount": 0,
      "HostWriteCommandCount": 0,
      "MediaAndDataIntegrityErrorCount": 0,
      "ErrorInformationLogEntryCount": 0
    }
  },
  "Capacity": {
    "Data": {
      "AllocatedBytes": 1099511627760,
      "ConsumedBytes": 1099511627760
    }
  },
  "CapacitySources": [
    {
      "@odata.type": "#Capacity.v1_2_0.CapacitySource",
      "Name": "Source1",
      "Id": "Source1",
      "ProvidedCapacity": {
        "Data": {
          "ConsumedBytes": 1099511627760,
          "AllocatedBytes": 1099511627760
        }
      }
    }
  ]
}
```

```
    },  
    "ProvidingDrives": {  
      "@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/EnduranceGroup1/CapacitySources/Source1/ProvidingDrives"  
    },  
    "ProvidingPools": {  
      "@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/EnduranceGroup1/CapacitySources/Source1/ProvidingPools"  
    },  
    "@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/EnduranceGroup1/CapacitySources/Source1"  
  }  
],  
"@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/EnduranceGroup1",  
"@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."  
}
```

6.6.2 Property Mapping

6.6.2.1 AllocatedPools

The mapping for `AllocatedPools` is summarized in Table 159.

Table 159: `AllocatedPools` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>AllocatedPools</code>	NVM Spec: Property / Field: NVMe 2.0: Identify Command / NVM Set List (CNS 04h) NVM Spec: Section: Figure NVMe 2.0: 5.17.2.4: Figure 278
Type	<code>StoragePoolCollection.StoragePoolCollection</code>	NVM Set List
Description	A reference to the collection of storage pools allocated from this storage pool.	A list of NVM sets
LongDescription	The value of this property shall contain a reference to the collection of storage pools allocated from this storage pool.	The data structure is an ordered list of up to 31 NVM Set Attribute Entry data structures, sorted by NVM Set Identifier, starting with the first NVM Set Identifier supported by the NVM subsystem that is equal to or greater than the NVM Set Identifier indicated in CDW11.NVMSETID and are accessible by the controller processing the command
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes	Contains a pointer to the NVM Set allocated from this Endurance Group.	

6.6.2.2 Capacity.Data.AllocatedBytes

The mapping for `Capacity.Data.AllocatedBytes` is summarized in Table 160.

Table 160: `Capacity.Data.AllocatedBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Capacity.Data.AllocatedBytes</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h) Total Endurance Group Capacity (TEGCAP) NVM Spec: Section: Figure NVMe 2.0: 5.116.1.10: Figure 218 Get Log Page – Endurance Group Information Log (Log Identifier 09h) : Total Endurance Group Capacity (TEGCAP) (bytes 175:160)
Type	Int64	16-Bytes
Description	The number of bytes currently allocated by the storage system in this data store for this data type.	This field indicates the total NVM capacity in this Endurance Group. The value is in bytes.
LongDescription	The value shall be the number of bytes currently allocated by the storage system in this data store for this data type.	This field indicates the total NVM capacity in this Endurance Group. The value is in bytes. If this field is cleared to 0h, the NVM subsystem does not report the total NVM capacity in this Endurance Group.
Mandatory	Optional Mandatory for NVMe Drives.	I/O Controller: Optional Admin Controller: Optional Discovery Controller: Prohibited
Notes		

6.6.2.3 Capacity.Data.ConsumedBytes

The mapping for `Capacity.Data.ConsumedBytes` is summarized in Table 161.

Table 161: `Capacity.Data.ConsumedBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Capacity.Data.ConsumedBytes</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h) Total Endurance Group Capacity (TEGCAP) - Unallocated Endurance Group Capacity (UEGCAP) NVM Spec: Section: Figure NVMe 2.0: 5.116.1.10: Figure 218 Get Log Page – Endurance Group Information Log (Log Identifier 09h) Subtract Unallocated Endurance Group Capacity (UEGCAP) (bytes 191:176) from Total Endurance Group Capacity (TEGCAP) (bytes 175:160)
Type	Int64	16-Bytes
Description	The number of bytes consumed in this data store for this data type.	Total Endurance Group Capacity (TEGCAP) - Unallocated Endurance Group Capacity (UEGCAP)
LongDescription	The value shall be the number of logical bytes currently consumed in this data store for this data type.	Total Endurance Group Capacity (TEGCAP) - Unallocated Endurance Group Capacity (UEGCAP)
Mandatory	Optional Mandatory for NVMe Drives.	I/O Controller: Optional Admin Controller: Optional Discovery Controller: Prohibited
Notes		Total Endurance Group Capacity (TEGCAP): This field indicates the total NVM capacity in this Endurance Group. The value is in bytes. If this field is cleared to 0h, the NVM subsystem does not report the total NVM capacity in this Endurance Group. Unallocated Endurance Group Capacity (UEGCAP): This field indicates the unallocated NVM capacity in this Endurance Group. The value is in bytes. If this field is cleared to 0h, the NVM subsystem does not report the unallocated NVM capacity in this Endurance Group.

6.6.2.4 CapacitySources

The mapping for `CapacitySources` is summarized in Table 162.

Table 162: CapacitySources mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CapacitySources	See notes
Type	Collection(Capacity.CapacitySource)	See Notes
Description	An array of space allocations to this volume.	A list of NVM Endurance Groups accessible by the controller and the Endurance Group Information Log for each Endurance Group.
LongDescription	Fully or partially consumed storage from a source resource. Each entry provides capacity allocation information from a named source resource.	See notes.
Mandatory	Required for NVMe Drives.	Optional
Notes	Contains the information about the providing capacity (e.g, NVMeSet) for this endurance group. Each entry in the collection is mapped such that the CapacitySource instance contains the overall capacity (in bytes) and the types and pointers to the underlying capacity sources.	For each entry in the CapacitySource collection, map the information from the NVMe specification: NVM Spec: Property / Field: NVMe 2.0: Identify Command / NVM Endurance Group List (CNS 19h) and NVMe 2.0: Endurance Group Information log (Log ID 09h) NVM Spec: Section :Figure NVMe 2.0: 5.17.2.18: Figure 288 and NVMe 2.0: 5.16.1.10: Figure 217. Identify command to access the Endurance Group List data structure (refer to section 5.17.2.18) to determine the Endurance Groups that are accessible by the controller. To determine the capacity information for each Endurance Group, the host uses the Get Log Page command to access the Endurance Group Information log page (refer to section 5.16.1.10).

6.6.2.5 CapacitySources@odata.count

The mapping for `CapacitySources@odata.count` is summarized in Table 163.

Table 163: `CapacitySources@odata.count` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>CapacitySources@odata.count</code>	NVM Spec: Property / Field: NVMe 2.0: Identify Command / NVM Endurance Group List (CNS 19h) NVM Spec: Section: Figure NVMe 2.0: 5.17.2.4: Figure 278. Bytes 00 (Number of Identifiers) in the NVM Endurance Group List
Type	(odata property)int64	Int64
Description	Count of the number of items in the <code>CapacitySources</code> array.	Number of Identifiers in the NVM Endurance Group List
LongDescription		This field contains the number of Endurance Group Identifiers in the list. There may be up to 2,047 identifiers in the list. If this field is cleared to 0h, then no Endurance Group Identifiers are in the list.
Mandatory	Mandatory	Optional
Notes		

6.6.2.6 Description The mapping for `Description` is summarized in Table 164.

Table 164: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Optional	
Notes	In Redfish, Description is a read-only field.	

6.6.2.7 Links.OwningStorageResource The mapping for `Links.OwningStorageResource` is summarized in Table 165.

Table 165: `Links.OwningStorageResource` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.OwningStorageResource</code>	N/A
Type	<code>Storage.Storage</code>	N/A
Description	A pointer to the Storage resource that owns or contains this StoragePool.	N/A
LongDescription	This shall be a pointer to the Storage resource that owns or contains this StoragePool.	N/A
Mandatory	Mandatory	
Notes	Contains a pointer to the NVM Subsystem that contains this Endurance Group.	A pointer to the owning subsystem - this is a Swordfish construct which points back to the subsystem

6.6.2.8 Name

The mapping for `Name` is summarized in Table 166.

Table 166: Name mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Name	NVM Spec: Property / Field: NVMe 2.0: Identify Command / CNS value of 19h (Endurance Group List) NVM Spec: Section: Figure NVMe 2.0: 5.17.2.18 Identify Controller data structure / CNS value of 19h to retrieve the Endurance Group List: Figure 288. This list will contain a Number of Endurance Group Identifiers equal to the Number of Identifiers (N) field in Bytes 01:00.
Type	String	2-Bytes
Description	The name of the resource or array member.	The list contains Endurance Group Identifiers of Endurance Groups that are accessible by the controller processing the command.
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	The Endurance Group List is a list of up to 2,047 Endurance Group Identifiers in increasing order that are accessible by the controller processing the command.
Mandatory	Mandatory	Optional (Mandatory for controllers that support Variable Capacity Management)
Notes	In Redfish, Name is a read-only field.	The Endurance Group Identifier is specified in the Log Specific Identifier field in Command Dword 11 of the Get Log Page command. Map the Endurance Group ID field to a string with the format: "0xABCD"

6.6.2.9 NVMeProperties.NVMePoolType The mapping for `NVMeProperties.NVMePoolType` is summarized in Table 167.

Table 167: `NVMeProperties.NVMePoolType`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	StoragePool.NVMePoolType	N/A
Type	Enum	N/A
Description	Indicates whether the StoragePool is used as an EnduranceGroup or an NVMeSet.	N/A
LongDescription	This property shall indicate whether the StoragePool is used as an EnduranceGroup or an NVMeSet.	N/A
Mandatory	Mandatory	
Notes	Set as "EnduranceGroup"	

6.6.2.10 NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent The mapping for `NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent` is summarized in Table 168.

Table 168: NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMe EnduranceGroupProperties.PredictedMediaLifeLeftPercent	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h) Inverse of the “Percentage Used” field (Byte 05) NVM Spec: Section: Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Decimal %	Decimal %
Description	The percentage of reads and writes that are predicted to be available for the media.	The Inverse of the “Percentage Used” field which contains a vendor specific estimate of the percentage of life used for the Endurance Group based on the actual usage and the manufacturer’s prediction of NVM life.
Long Description	This property shall contain an indicator of the percentage of life remaining in the drive’s media.	The Inverse of the “Percentage Used” field which contains a vendor specific estimate of the percentage of life used for the Endurance Group based on the actual usage and the manufacturer’s prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the Endurance Group has been consumed, but may not indicate an NVM failure.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: Optional Discovery Controller: Prohibited
Notes		The value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255. This value shall be updated once per power-on hour when the controller is not in a sleep state.

6.6.2.11 NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed` is summarized in Table 169.

Table 169: `NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 0 9h) “Percentage Used” field (Byte 05) NVM Spec: Section: Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	Decimal % from 00% - 255%
Description	A vendor-specific estimate of the percent life used for the endurance group based on the actual usage and the manufacturer prediction of NVM life.	Contains a vendor specific estimate of the percentage of life used for the Endurance Group based on the actual usage and the manufacturer’s prediction of NVM life.
LongDescription	This property shall contain A vendor-specific estimate of the percent life used for the endurance group based on the actual usage and the manufacturer prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the Endurance Group has been consumed, but may not indicate an NVM failure. According to the NVMe and JEDEC specs, the value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255.	Contains a vendor specific estimate of the percentage of life used for the Endurance Group based on the actual usage and the manufacturer’s prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the Endurance Group has been consumed, but may not indicate an NVM failure. The value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255. This value shall be updated once per power-on hour when the controller is not in a sleep state.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: Optional Discovery Controller: Prohibited
Notes		The value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255. This value shall be updated once per power-on hour when the controller is not in a sleep state. This value shall be scaled to a value between 0% - 100%. Calculate resulting value as 100 - value reported (PercentageUsed).

6.6.2.12 NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate` is summarized in Table 170.

Table 170: `NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Endurance Estimate” field (Bytes 32-47) NVM Spec: Section: NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	This property contains an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group assuming a write amplification of 1.	This field is an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group.
LongDescription	This property shall contain an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group assuming a write amplification of 1. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates endurance estimates are unsupported.	This field is an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group assuming a write amplification of 1 (i.e., no increase in the number of write operations performed by the device beyond the number of write operations requested by a host).
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited

Redfish/Swordfish	NVMe / NVMe-oF
Notes	<p>This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes written) and is rounded up (e.g., one indicates the number of bytes written is from 1 to 1,000,000,000, three indicates the number of bytes written is from 2,000,000,001 to 3,000,000,000). A value of 0h indicates that the controller does not report an Endurance Estimate. This value shall be scaled to a value between 0% - 100%. Calculate resulting value as 100 - value reported (PercentageUsed).</p>

6.6.2.13 NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead` is summarized in Table 171.

Table 171: `NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Data Units Read” field (Bytes 48 - 63) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
	Int64	16-Bytes
Description	The property contains the total number of data units read from this endurance group.	Contains the total number of data bytes that have been read from the Endurance Group.
LongDescription	The property shall contain the total number of data units read from this endurance group. This value does not include controller reads due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.	Contains the total number of data bytes that have been read from the Endurance Group. This value does not include controller reads due to internal operations such as garbage collection.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: Optional Discovery Controller: Prohibited
Notes		This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes read) and is rounded up (e.g., one indicates the number of bytes read is from 1 to 1,000,000,000, three indicates the number of bytes read is from 2,000,000,001 to 3,000,000,000). A value of 0h indicates that the controller does not report the number of Data Units Read

6.6.2.14 NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten` is summarized in Table 172.

Table 172: NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Data Units Written” field (Bytes 64 - 79) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	The property contains the total number of data units written from this endurance group.	Contains the total number of data bytes that have been written to the Endurance Group.
LongDescription	The property shall contain the total number of data units written from this endurance group. This value does not include controller writes due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.	Contains the total number of data bytes that have been written to the Endurance Group. This value does not include controller writes due to internal operations such as garbage collection.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes		This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes written) and is rounded up (e.g., one indicates the number of bytes written is from 1 to 1,000,000,000, three indicates the number of bytes written is from 2,000,000,001 to 3,000,000,000). A value of 0h indicates that the controller does not report the number of Data Units Written.

6.6.2.15 NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten` is summarized in Table 173.

Table 173: `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Media Units Written” field (Bytes 80 - 95) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	The property contains the total number of data units written from this endurance group.	Contains the total number of data bytes that have been written to the Endurance Group including both host and controller writes (e.g., garbage collection).
LongDescription	The property shall contain the total number of data units written from this endurance group. This value includes host and controller writes due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.	Contains the total number of data bytes that have been written to the Endurance Group including both host and controller writes (e.g., garbage collection).
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes		This value is reported in billions (i.e., a value of 1 corresponds to 1,000,000,000 bytes written) and is rounded up (e.g., one indicates the number of bytes written is from 1 to 1,000,000,000, three indicates the number of bytes written is from 2,000,000,001 to 3,000,000,000). A value of 0h indicates that controller does not report the number of Media Units Written.

6.6.2.16 NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount` is summarized in Table 174.

Table 174: NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Host Read Commands” field (Bytes 96 - 111) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	This property contains the number of read commands completed by all controllers in the NVM subsystem for the Endurance Group.	Contains the number of User Data Read Access Commands completed by the controller
LongDescription	This property shall contain the number of read commands completed by all controllers in the NVM subsystem for the Endurance Group. For the NVM command set, the is the number of compare commands and read commands.	Contains the number of User Data Read Access Commands completed by the controller
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes		Refer to the specific NVMe I/O Command Set specification for the list of User Data Read Access Commands that affect this field.

6.6.2.17 NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount` is summarized in Table 6.6.2.17.

Table 175: NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Host Write Commands” field (Bytes 112 - 127) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	This property contains the number of write commands completed by all controllers in the NVM subsystem for the Endurance Group.	Contains the number of User Data Out Commands completed by the controller.
LongDescription	This property shall contain the number of write commands completed by all controllers in the NVM subsystem for the Endurance Group. For the NVM command set, this is the number of compare commands and write commands.	Contains the number of User Data Out Commands completed by the controller.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes		Refer to the specific I/O Command Set specification for the list of User Data Out Commands that affect this field.

Table: NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount mapping

6.6.2.18 NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount` is summarized in Table 176.

Table 176: NVMeEnduranceGroupProperties.EndGrpLifetime MediaAndDataIntegrityErrorCount mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Media and Data Integrity Errors” field (Bytes 128 - 143) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	This property contains the number of occurrences where the controller detected an unrecovered data integrity error for the Endurance Group.	Contains the number of occurrences where the controller detected an unrecovered data integrity error.
LongDescription	This property shall contain the number of occurrences where the controller detected an unrecovered data integrity error for the Endurance Group. Errors such as uncorrectable ECC, CRC checksum failure, or LBA tag mismatch are included in this field.	Contains the number of occurrences where the controller detected an unrecovered data integrity error.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes		Errors such as uncorrectable ECC, CRC checksum failure, or LBA tag mismatch are included in this field. Errors introduced as a result of a Write Uncorrectable command (refer to the NVM Command Set specification) may or may not be included in this field.

6.6.2.19 NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount` is summarized in Table 177.

Table 177: `NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount</code>	NVM Spec: Property / Field: NVMe 2.0: Endurance Group Information Log (Log Identifier 09h)“Number of Error Information Log Entries” field (Bytes 144 - 159) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217
Type	Int64	16-Bytes
Description	This property contains the number of error information log entries over the life of the controller for the endurance group.	Contains the number of Error Information log entries over the life of the controller.
LongDescription	This property shall contain the number of error information log entries over the life of the controller for the endurance group.	Contains the number of Error Information log entries over the life of the controller.
Mandatory	Mandatory	I/O Controller: Optional Admin Controller: OptionalDiscovery Controller: Prohibited
Notes		

6.6.2.20 NVMeSetProperties.SetIdentifier The mapping for `NVMeSetProperties.SetIdentifier` is summarized in Table 178.**Table 178:** `NVMeSetProperties.SetIdentifier`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMe SetProperties.SetIdentifier</code>	NVM Spec: Property / Field: NVMe 2.0: Identify Command / NVM Set List (CNS 04h) NVM Spec: Section:Figure NVMe 2.0: 5.17.2.4
Type	String	16-bits
Description	A 16-bit hex value that contains the NVMe Set identifier.	This field indicates the identifier of the NVM Set in the NVM subsystem that is described by this entry.
LongDescription	This property shall contain a 16-bit hex value that contains the NVMe Set identifier. The NVM Set identifier is unique within a subsystem. Reserved values include 0.	This field indicates the identifier of the NVM Set in the NVM subsystem that is described by this entry. Identify Command / CNS value of 04h to retrieve the NVM Set List. The NVM Set Identifier is retrieved in bytes 00:01 of each NVM Set Attributes entry in the NVM Set List
Mandatory	Do Not Implement	Optional
Notes		

6.6.2.21 NVMeSetProperties.OptimalWriteSizeBytes The mapping for `NVMeSetProperties.OptimalWriteSizeBytes` is summarized in Table 179.

Table 179: `NVMeSetProperties.OptimalWriteSizeBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.OptimalWriteSizeBytes</code>	NVM Spec: Property / Field: NVMe 2.0: Identify Command / NVM Set List (CNS 04h) f/ Optimal Write Size NVM Spec: Section: Figure NVMe 2.0: 5.17.2.4 Identify Command / NVM Set List (CNS value of 04h to retrieve the NVM Set List): Figure 278. The Optimal Write Size is retrieved in bytes 12:15 of each NVM Set Attributes entry (Figure 279) in the NVM Set List
Type	Int64	Bytes
Description	This property contains the Optimal Write Size in Bytes for this NVMe Set.	This field indicates the size in bytes for optimal write performance.
LongDescription	This property shall contain the Optimal Write Size in Bytes for this NVMe Set.	This field indicates the size in bytes for optimal write performance. A value of 0h indicates that no Optimal Write Size is specified. This field should be cleared to 0h when namespaces within an NVM Set have different User Data Formats that do not allow an Optimal Write Size to be specified.
Mandatory	Do Not Implement	Optional
Notes		

6.6.2.22 NVMeSetProperties.EnduranceGroupIdentifier The mapping for `NVMeSetProperties.EnduranceGroupIdentifier` is summarized in Table 180.

Table 180: NVMeSetProperties.EnduranceGroupIdentifier mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeSetProperties.EnduranceGroupIdentifier	N/A
Type	String	N/A
Description	A 16-bit hex value that contains the endurance group identifier.	N/A
LongDescription	This property shall contain a 16-bit hex value that contains the endurance group identifier. The endurance group identifier is unique within a subsystem. Reserved values include 0.	N/A
Mandatory	Do Not Implement	
Notes		Do not implement NVMeSetProperties as part of an EnduranceGroup.

6.6.2.23 NVMeSetProperties.Random4kReadTypicalNanoSeconds The mapping for `NVMeSetProperties.Random4kReadTypicalNanoSeconds` is summarized in Table 181.

Table 181: `NVMeSetProperties.Random4kReadTypicalNanoSeconds` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.Random4kReadTypicalNanoSeconds</code>	N/A
Type	Int64	N/A
Description	Indicates the typical time to complete a 4k read in 100 Nano-second units when the NVM Set is in a Predictable Latency Mode Deterministic Window and there is 1 outstanding command per NVM Set.	N/A
LongDescription	This property shall contain the typical time to complete a 4k read in 100 nano-second units when the NVM Set is in a Predictable Latency Mode Deterministic Window and there is 1 outstanding command per NVM Set.	N/A
Mandatory	Do Not Implement	
Notes	Do not implement <code>NVMeSetProperties</code> as part of an <code>EnduranceGroup</code> .	

6.6.2.24 Status.Health

The mapping for `Status.Health` is summarized in Table 182.

Table 182: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	NVM Spec: Property / Field: NVMe 2.0: Get Log Page – Endurance Group Information Log (Log Identifier 09h) / Critical Warning NVM Spec: Section: Figure: NVMe 2.0: 5.16.1.10: Figure 217 / Critical Warning (bytes 00)
Type	Resource.Health	Bit value
Description	The health state of this resource in the absence of its dependent resources.	This field indicates critical warnings for the state of the Endurance Group.
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	This field indicates critical warnings for the state of the Endurance Group. Each bit corresponds to a critical warning type; multiple bits may be set to '1'. If a bit is cleared to '0', then that critical warning does not apply. Critical warnings may result in an asynchronous event notification to the host. Bits in this field represent the current associated state and are not persistent.
Mandatory	Optional	I/O Controller: Optional Admin Controller: Optional Discovery Controller: Prohibited
Notes	Possible Values: OK / Warning / Critical	If bit 3 is set to '1', then all namespaces in the Endurance Group have been placed in read only mode for reasons other than a change in the write protect state of the namespace. The controller shall not set this bit to '1' if the read-only condition on the Endurance Group is a result of a change in the write protection state of all namespaces in the Endurance Group. If bit 2 is set to '1', then the Endurance Group reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability. If bit 0 is set to '1', then the available spare capacity of the Endurance Group has fallen below the threshold.

6.6.2.25 Status.State The mapping for `Status.State` is summarized in Table 183.

Table 183: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	N/A
Type	Resource.State (enum)	N/A
Description	The known state of the resource, such as, enabled.	N/A
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	N/A
Mandatory	Do Not Implement	Do not implement
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySare / InTest / Starting / ABsent / UnavaialableOffline / Deferring / Quiesced / Updating / Qualified	There is not a clear mapping for State of an Endurance Group. Do not implement this property.

6.7 NVM Set

The Redfish/Swordfish Storage Pool schema is used to represent an NVM Set.

6.7.1 Mockup

The following mockup shows a sample representation of the Storage Pool schema used to represent an NVM Set.

```
{
  "@odata.type": "#StoragePool.v1_7_1.StoragePool",
  "Id": "1",
  "Name": "Set 1",
  "Description": "First Set",
  "NVMeProperties": {
    "NVMePoolType": "NVMeSet"
  },
  "NVMeSetProperties": {
    "SetIdentifier": "0x1F",
    "EnduranceGroupIdentifier": "0x1",
    "Random4kReadTypicalNanoSeconds": 34534345348,
    "UnallocatedNVMeNamespaceCapacityBytes": 5497558138880,
    "OptimalWriteSizeBytes": 512
  },
  "Capacity": {
    "Data": {
      "AllocatedBytes": 10995116277760,
      "ConsumedBytes": 5497558138880
    }
  },
  "CapacitySources": [{
    "@odata.type": "#Capacity.v1_2_0.CapacitySource",
    "Name": "Source1",
    "Id": "Source1",
    "Description": "Capacity Source for NVM Set",
    "ProvidedCapacity": {
      "Data": {
        "AllocatedBytes": 10737418240
      }
    },
    "ProvidingPools": {
      "@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/EnduranceGroup1/CapacitySources/Source1/ProvidingPools"
    },
    "@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/NVMeSet1/CapacitySources/Source1"
  }],
  "AllocatedVolumes": {
    "@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/NVMeSet1/AllocatedVolumes"
  }
}
```

```
},  
"@odata.id": "/redfish/v1/Storage/FabricAttachArray/StoragePools/NVMeSet1",  
"@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."  
}
```

6.7.2 Property Mapping

6.7.2.1 AllocatedVolumes The mapping for `AllocatedVolumes` is summarized in Table 184.

Table 184: `AllocatedVolumes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>AllocatedVolumes</code>	N/A
Type	<code>VolumeCollection.VolumeCollection</code>	N/A
Description	A reference to the collection of volumes allocated from this storage pool.	N/A
LongDescription	The value of this property shall contain a reference to the collection of volumes allocated from this storage pool.	N/A
Mandatory	Mandatory	
Notes	A pointer to the set of namespaces allocated from this NVM Set.	The allocated volumes contains pointers to the allocated volumes objects. These are the set of namespaces created from this NVM Set.

6.7.2.2 Capacity.Data.AllocatedBytes

The mapping for `Capacity.Data.AllocatedBytes` is summarized in Table 185

Table 185: `Capacity.Data.AllocatedBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Capacity.Data.AllocatedBytes</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / Total NVM Set Capacity NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 16-31 (Total NVM Set Capacity)
Type	Int64	16 Bytes
Description	The number of bytes currently allocated by the storage system in this data store for this data type.	This field indicates the total NVM capacity in this NVM Set.
LongDescription	The value shall be the number of bytes currently allocated by the storage system in this data store for this data type.	This value shall be the total NVM capacity in this NVM Set. The value is in bytes.
Mandatory	Mandatory	Mandatory
Notes		

6.7.2.3 Capacity.Data.ConsumedBytes

The mapping for `Capacity.Data.ConsumedBytes` is summarized in Table 186.

Table 186: `Capacity.Data.ConsumedBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Capacity.Data.ConsumedBytes</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / Unallocated NVM Set Capacity NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 32 - 47 (Unallocated NVM Set Capacity)
Type	Int64	16 Bytes
Description	The maximum number of bytes that can be allocated in this data store for this data type.	This field indicates the unallocated NVM capacity in this NVM Set.
LongDescription	The value shall be the maximum number of bytes that can be allocated in this data store for this data type.	This field indicates the unallocated NVM capacity in this NVM Set. The value is in bytes.
Mandatory	Mandatory	Optional
Notes		This is calculated as “Total NVM Set Capacity” - “Unallocated NVM Set Capacity”.

6.7.2.4 CapacitySources

The mapping for `CapacitySources` is summarized in Table 187.

Table 187: CapacitySources mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CapacitySources	N/A
Type	Collection(Capacity.CapacitySource)	N/A
Description	An array of space allocations to this volume.	N/A
LongDescription	Fully or partially consumed storage from a source resource. Each entry provides capacity allocation information from a named source resource.	N/A
Mandatory	Optional Recommended to not implement for NVMe Drives.	DNI
Notes	Contains the information about the providing capacity (e.g, memory) for this namespace.	

6.7.2.5 CapacitySources@odata.count The mapping for `CapacitySources@odata.count` is summarized in Table 188.

Table 188: `CapacitySources@odata.count` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>CapacitySources@odata.count</code>	N/A
Type	(odata property)int64	N/A
Description	Count of the number of items in the <code>CapacitySources</code> array.	N/A
LongDescription		N/A
Mandatory	Optional Do Not Implement for NVMe Drives.	DNI
Notes		

6.7.2.6 Description The mapping for `Description` is summarized in Table 189.

Table 189: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	See note below.
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Optional	
Notes	In Redfish, Description is a read-only field.	Return the common description: “An NVM Set is a collection of NVM that is separate (logically and potentially physically) from NVM in other NVM Sets. One or more namespaces may be created within an NVM Set and those namespaces inherit the attributes of the NVM Set. A namespace is wholly contained within a single NVM Set and shall not span more than one NVM Set.”

6.7.2.7 Links.OwningStorageResource The mapping for `Links.OwningStorageResource` is summarized in Table 190.

Table 190: `Links.OwningStorageResource` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.OwningStorageResource</code>	N/A
Type	<code>Storage.Storage</code>	N/A
Description	A pointer to the Storage resource that owns or contains this StoragePool.	N/A
LongDescription	This shall be a pointer to the Storage resource that owns or contains this StoragePool.	N/A
Mandatory	Mandatory	
Notes	Contains a pointer to the NVM Subsystem that contains this NVM Set.	

6.7.2.8 Name

The mapping for `Name` is summarized in Table 191

Table 191: Name mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Name	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / NVM Set Identifier NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 00 - 01 (NVM Set Identifier)
Type	String	16-bit value
Description	The name of the resource or array member.	This field specifies the NVM Set ID
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	This field specifies the NVM Set ID. An NVM Set Identifier is a 16-bit value that specifies the NVM Set
Mandatory	Mandatory	
Notes	In Redfish, Name is a read-only field.	Map the NVMSETID field to a string with the format: "0xABCD"

6.7.2.9 NVMeProperties.NVMePoolType The mapping for `NVMeProperties.NVMePoolType` is summarized in Table 192.

Table 192: `NVMeProperties.NVMePoolType`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	StoragePool.NVMePoolType	N/A
Type	Enum	N/A
Description	Indicates whether the StoragePool is used as an EnduranceGroup or an NVMeSet.	N/A
LongDescription	This property shall indicate whether the StoragePool is used as an EnduranceGroup or an NVMeSet.	N/A
Mandatory	Mandatory	
Notes	Set as "NVMeSet"	

6.7.2.10 NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent The mapping for `NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent` is summarized in Table 193.

Table 193: `NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.PredictedMediaLifeLeftPercent</code>	N/A
Type	Decimal %	N/A
Description	The percentage of reads and writes that are predicted to be available for the media.	N/A
LongDescription	This property shall contain an indicator of the percentage of life remaining in the drive's media.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.11 NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed` is summarized in Table 194.

Table 194: `NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.PercentUsed</code>	N/A
Type	Int64	N/A
Description	A vendor-specific estimate of the percent life used for the endurance group based on the actual usage and the manufacturer prediction of NVM life.	N/A
LongDescription	This property shall contain A vendor-specific estimate of the percent life used for the endurance group based on the actual usage and the manufacturer prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the Endurance Group has been consumed, but may not indicate an NVM failure. According to the NVMe and JEDEC specs, the value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.12 NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate` is summarized in Table 195.

Table 195: `NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.EnduranceEstimate</code>	N/A
Type	Int64	N/A
Description	This property contains an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group assuming a write amplification of 1.	N/A
LongDescription	This property shall contain an estimate of the total number of data bytes that may be written to the Endurance Group over the lifetime of the Endurance Group assuming a write amplification of 1. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates endurance estimates are unsupported.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.13 NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead` is summarized in Table 196.

Table 196: `NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsRead</code>	N/A
Type	Int64	N/A
Description	The property contains the total number of data units read from this endurance group.	N/A
LongDescription	The property shall contain the total number of data units read from this endurance group. This value does not include controller reads due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.14 NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten` is summarized in Table 197.

Table 197: NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.DataUnitsWritten	N/A
Type	Int64	N/A
Description	The property contains the total number of data units written from this endurance group.	N/A
LongDescription	The property shall contain the total number of data units written from this endurance group. This value does not include controller writes due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.15 NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten` is summarized in Table 198.

Table 198: `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.MediaUnitsWritten</code>	N/A
Type	Int64	N/A
Description	The property contains the total number of data units written from this endurance group.	N/A
LongDescription	The property shall contain the total number of data units written from this endurance group. This value includes host and controller writes due to internal operations such as garbage collection. The value is reported in billions, where a value of 1 corresponds to 1 billion bytes written, and is rounded up. A value of zero indicates the property is unsupported.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.16 NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount` is summarized in Table 199.

Table 199: NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.HostReadCommandCount	N/A
Type	Int64	N/A
Description	This property contains the number of read commands completed by all controllers in the NVM subsystem for the Endurance Group.	N/A
LongDescription	This property shall contain the number of read commands completed by all controllers in the NVM subsystem for the Endurance Group. For the NVM command set, this is the number of compare commands and read commands.	N/A
Mandatory	Do Not Implement	DNI

6.7.2.17 NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount` is summarized in Table 200.

Table 200: NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NVMeEnduranceGroupProperties.EndGrpLifetime.HostWriteCommandCount	N/A
Type	Int64	N/A
Description	This property contains the number of write commands completed by all controllers in the NVM subsystem for the Endurance Group.	N/A
LongDescription	This property shall contain the number of write commands completed by all controllers in the NVM subsystem for the Endurance Group. For the NVM command set, this is the number of compare commands and write commands.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.18 NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount` is summarized in Table 201.

Table 201: `NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.MediaAndDataIntegrityErrorCount</code>	N/A
Type	Int64	N/A
Description	This property contains the number of occurrences where the controller detected an unrecovered data integrity error for the Endurance Group.	N/A
LongDescription	This property shall contain the number of occurrences where the controller detected an unrecovered data integrity error for the Endurance Group. Errors such as uncorrectable ECC, CRC checksum failure, or LBA tag mismatch are included in this field.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.19 NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount The mapping for `NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount` is summarized in Table 202.

Table 202: `NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeEnduranceGroupProperties.EndGrpLifetime.ErrorInformationLogEntryCount</code>	N/A
Type	Int64	N/A
Description	This property contains the number of error information log entries over the life of the controller for the endurance group.	N/A
LongDescription	This property shall contain the number of error information log entries over the life of the controller for the endurance group.	N/A
Mandatory	Do Not Implement	DNI
Notes		

6.7.2.20 NVMeSetProperties.SetIdentifier The mapping for `NVMeSetProperties.SetIdentifier` is summarized in Table 203.

Table 203: `NVMeSetProperties.SetIdentifier`

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.SetIdentifier</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / NVM Set Identifier NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 00 - 01 (NVM Set Identifier)
Type	String	16 Bit value
Description	A 16-bit hex value that contains the NVMe Set identifier.	This field specifies the NVM Set ID.

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This property shall contain a 16-bit hex value that contains the NVMe Set identifier. The NVM Set identifier is unique within a subsystem. Reserved values include 0.	This field specifies the NVM Set ID. An NVM Set Identifier is a 16-bit value that specifies the NVM Set
Mandatory	Mandatory	Mandatory
Notes		Return as hex value as described in the Swordfish schema.

6.7.2.21 NVMeSetProperties.OptimalWriteSizeBytes The mapping for `NVMeSetProperties.OptimalWriteSizeBytes` is summarized in Table 204.

Table 204: `NVMeSetProperties.OptimalWriteSizeBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.OptimalWriteSizeBytes</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / Optimal Write Size NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 12 - 15 (Optimal Write Size)
Type	Int64	4-Bytes
Description	This property contains the Optimal Write Size in Bytes for this NVMe Set.	This field indicates the size in bytes for optimal write performance.
LongDescription	This property shall contain the Optimal Write Size in Bytes for this NVMe Set.	This field indicates the size in bytes for optimal write performance. A value of 0h indicates that no Optimal Write Size is specified. This field should be cleared to 0h when namespaces within an NVM Set have different User Data Formats that do not allow an Optimal Write Size to be specified.
Mandatory	Mandatory	Mandatory
Notes		

6.7.2.22 NVMeSetProperties.EnduranceGroupIdentifier The mapping for `NVMeSetProperties.EnduranceGroupIdentifier` is summarized in Table 205.

Table 205: `NVMeSetProperties.EnduranceGroupIdentifier` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.EnduranceGroupIdentifier</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / Endurance Group Identifier NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 02 - 03 (Endurance Group Identifier)
Type	String	2iBytes
Description	A 16-bit hex value that contains the endurance group identifier.	This field indicates the Endurance Group for this NVM Set.
LongDescription	This property shall contain a 16-bit hex value that contains the endurance group identifier. The endurance group identifier is unique within a subsystem. Reserved values include 0.	This field indicates the Endurance Group for this NVM Set.
Mandatory	Mandatory	Mandatory
Notes		

6.7.2.23 NVMeSetProperties.Random4kReadTypicalNanoSeconds The mapping for `NVMeSetProperties.Random4kReadTypicalNanoSeconds` is summarized in Table 206.

Table 206: `NVMeSetProperties.Random4kReadTypicalNanoSeconds` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.Random4kReadTypicalNanoSeconds</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / Random 4 KiB Read Typical NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 08 - 11 (Random 4 KiB Read Typical)
Type	Int64	4 Bytes
Description	Indicates the typical time to complete a 4k read in 100 nano-second units when the NVM Set is in a Predictable Latency Mode Deterministic Window and there is 1 outstanding command per NVM Set.	This field indicates the typical time to complete a 4 KiB random read in 100 nanosecond units
LongDescription	This property shall contain the typical time to complete a 4k read in 100 nano-second units when the NVM Set is in a Predictable Latency Mode Deterministic Window and there is 1 outstanding command per NVM Set.	This field indicates the typical time to complete a 4 KiB random read in 100 nanosecond units when the NVM Set is in a Predictable Latency Mode Deterministic Window and there is 1 outstanding command per NVM Set
Mandatory	Mandatory	
Notes		Convert from 100 nanosecond units to nanosecond units.

6.7.2.24 NVMeSetProperties.UnallocatedNVMNamespaceCapacityBytes The mapping for `NVMeSetProperties.UnallocatedNVMNamespaceCapacityBytes` is summarized in Table 207.

Table 207: `NVMeSetProperties.UnallocatedNVMNamespaceCapacityBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>NVMeSetProperties.UnallocatedNVMNamespaceCapacityBytes</code>	NVM Spec Property / Field: Identify Command / NVM Set List (CNS 04h) / NVM Set Attributes Entry / Unallocated NVM Set Capacity NVM Spec: Section:Figure NVMe 2.0: Section 5.15.2.4, Figure 278: NVM Set Attributes List / Figure 279: NVM Set Attributes Entry: Bytes 32 - 47 (Unallocated NVM Set Capacity)
Type	Int64	Int 64
Description	Indicates the unallocated capacity of the NVMe Set in bytes.	This field indicates the unallocated NVM capacity in this NVM Set.
LongDescription	This property shall contain the unallocated capacity of the NVMe Set in bytes.	This field indicates the unallocated NVM capacity in this NVM Set. The value is in bytes.
Mandatory	Mandatory	
Notes		

6.7.2.25 Status.State The mapping for `Status.State` is summarized in Table 208.

Table 208: `Status.State` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Status.State</code>	N/A
Type	Resource.State (enum)	N/A
Description	The known state of the resource, such as, enabled.	N/A

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but it can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	N/A
Mandatory	Optional	DNI
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySpare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified	

6.7.2.26 Status.Health The mapping for `Status.Health` is summarized in Table 209.

Table 209: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	N/A
Type	Resource.Health	N/A
Description	The health state of this resource in the absence of its dependent resources.	N/A
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	N/A
Mandatory	Optional	DNI
Notes	Possible Values: OK / Warning / Critical	

6.8 Drive

The Redfish/Swordfish Drive schema represent the physical attributes of an NVMe drive object.

6.8.1 Mockup

The following mockup shows a sample representation of the Volume schema used to represent an NVM Namespace.

```
{
  "@odata.type": "#Drive.v1_17_0.Drive",
  "Id": "0THGR0KP",
  "Name": "Drive 1",
  "Status": {
    "State": "Enabled",
    "Health": "OK"
  },
  "StatusIndicator": "OK",
  "CapacityBytes": 300067890136,
  "FailurePredicted": false,
  "Protocol": "SAS",
  "MediaType": "HDD",
  "Manufacturer": "HDD-Company",
  "SerialNumber": "0THGR0KP",
  "PartNumber": "HUC156030CSS200",
  "Identifiers": [
    {
      "DurableNameFormat": "NAA",
      "DurableName": "300062B202B21849"
    }
  ],
  "RotationSpeedRPM": 15000,
  "BlockSizeBytes": 512,
  "NegotiatedSpeedGbs": 12,
  "Metrics": {
    "@odata.id": "/redfish/v1/Chassis/StorageEnclosure1/Drives/0THGR0KP/Metrics"
  },
  "Links": {
    "Volumes": [
      {
        "@odata.id": "/redfish/v1/Systems/Sys-1/Storage/DirectAttachStorageSystem/Volumes/Volume1"
      }
    ],
    "StoragePools": [
      {
        "@odata.id": "/redfish/v1/Systems/Sys-1/Storage/DirectAttachStorageSystem/StoragePools/SimpleSystemPool"
      }
    ]
  }
}
```

```
    ]  
  },  
  "@odata.id": "/redfish/v1/Chassis/StorageEnclosure1/Drives/0THGR0KP",  
  "@Redfish.Copyright": "Copyright 2015-2023 SNIA. All rights reserved."  
}
```

6.8.2 Property Mapping

6.8.2.1 Actions.#Drive.Reset The mapping for `Actions.#Drive.Reset` is summarized in Table 210.

Table 210: `Actions.#Drive.Reset` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Actions.#Drive.Reset</code>	NVM Spec Property / Field: Set Property Command / NSSR: NVM Subsystem Reset (Controller Property Offset 20h) NSSD: NVM Subsystem Shutdown (Controller Property Offset 64h) NVM Spec: Section:Figure NVMe 2.0: Section 3.1.3: Figure 35: Offset 20h and 64h Further information on NVM Subsystem Reset Control (NSSRC): NVMe 2.0: Section 3.1.3.7: Figure 48 and NVM Subsystem Shutdown Control (NSSD): NVMe 2.0: Section 3.1.3.20: Figure 61
Type	Action (Special form of POST)	NVMe Administrative command
Description	This action resets this drive.	This is used to initiate a controller reset or shutdown depending on the control offset used.

	Redfish/Swordfish	NVMe / NVMe-oF
LongDescription	This action shall reset this drive.	NVM Subsystem Reset Control (NSSRC): A write of the value 4E564D65h (“NVMe”) to this field initiates an NVM Subsystem Reset. NVM Subsystem Shutdown Control (NSSC): A write of the value 4E726D6Ch (“Nrml”) to this field initiates a normal NVM Subsystem Shutdown on every controller (in the domain associated with the controller when CAP.CPS is set to 10b as specified in section 3.6.3.1; or <ul style="list-style-type: none"> in the NVM subsystem when CAP.CPS is set to 11b in the NVM subsystem as specified in section 3.6.3.2. A write of the value 41627074h (“Abpt”) to this field initiates an abrupt NVM subsystem shutdown on every controller: <ul style="list-style-type: none"> in the domain associated with the controller when CAP.CPS is set to 10b as specified in section 3.6.3.1; in the NVM subsystem when CAP.CPS is set to 11b in the NVM subsystem
Mandatory	Optional Mandatory for NVMe Drives	Optional
Notes	This action has a mandatory property of “ResetType”, which can be any of On/ForceOff/GracefulShutdown/GracefulRestart/Nmi/ForceRestart/ForceOn/PowerCycle.	Usage: A normal NVM Subsystem shutdown maps to GracefulShutdown; Subsystem Reset maps to ForceRestart; abrupt Subsystem Shutdown maps to ForceOff. If an implementation that supports the functionality, then they may implement PowerCycle.

6.8.2.2 Actions.#Drive.SecureErase The mapping for `Actions.#Drive.SecureErase` is summarized in Table 211.

Table 211: Actions.#Drive.SecureErase mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Actions.#Drive.SecureErase	NVM Spec Property / Field: Sanitize Operation NVM Spec: Section:Figure NVM Base Specification 2.0: 5.24 Sanitize Command NVM Base Specification 2.0: Section 8.21 Sanitize Operations NVM Base Specification 2.0: Annex A “Sanitize Operation Considerations (Informative)”
Type	Action (Special form of POST)	NVMe Administrative command
Description	This action securely erases the contents of the drive.	The sanitize administrative command operation makes all user data previously written to the device inaccessible.
LongDescription	This action shall securely erase the drive.	The Sanitize command is used to start a sanitize operation or to recover from a previously failed sanitize operation. All user data in the NVM subsystem is altered such that recovery of the previous user data from any cache or the non-volatile media is not possible.
Mandatory	Mandatory	Optional
Notes	The action parameter SanitizationType supported are: BlockErase, CryptographicErase, and Overwrite. When Overwrite is used, OverwritePasses must also be specified.	Maps to sanitize. Implementation can support any variant. The sanitize operation types that may be supported are Block Erase, Crypto Erase, and Overwrite. All sanitize operations are processed in the background (i.e., completion of the Sanitize command does not indicate completion of the sanitize operation)

6.8.2.3 Assembly.BinaryDataURI

The mapping for `Assembly.BinaryDataURI` is summarized in Table 212.

Table 212: `Assembly.BinaryDataURI` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Assembly.BinaryDataURI</code>	N/A
Type	String	N/A
Description	The URI at which to access an image of the assembly information.	N/A
LongDescription	This property shall contain the URI at which to access an image of the assembly information, using the Redfish protocol and authentication methods. The Service provides this URI for the download of the OEM-specific binary image of the assembly data. An HTTP GET from this URI shall return a response payload of MIME type <code>application/octet-stream</code> . If the service supports it, an HTTP PUT to this URI shall replace the binary image of the assembly.	N/A
Mandatory	Recommended	
Notes		N/A for NVMe 2.0

6.8.2.4 BlockSizeBytes

The mapping for `BlockSizeBytes` is summarized in Table 213.

Table 213: `BlockSizeBytes` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>BlockSizeBytes</code>	<p>NVM Spec Property / Field: Formatted LBA Size (FLBAS) retrieved in some Identify Namespace data structures for the specified NSID or the common namespace capabilities for the NVM Command Set (CNS 00h). Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved NVM Spec:</p> <p>Section:Figure NVMe NVM Command Set Specification 1.0b: Section 4.1.5.1: Figure 97: Bytes 26 - Formatted LBA Size (FLBAS) and uses the Number LBA Formats (NBLAF) field defined in Section 4.1.5.1: Figure 97: Bytes 25 (Number of LBA Formats) NVMe Express NVMe Zoned Namespace Command Set Specification 1.1b: Section A.5: Figure 53 (Size and Capacity Fields): Bytes 26.</p>
Type	Int64	Int-64
Description	The size, in bytes, of the smallest addressable unit, or block.	The LBA data size & metadata size combination that the namespace has been formatted with.
LongDescription	This property shall contain size of the smallest addressable unit of the associated drive or device.	This field indicates the LBA data size & metadata size combination that the namespace has been formatted with.
Mandatory	Mandatory	Only Mandatory / applicable for the NVM Command Set and I/O Command Sets based on the NVM Command Set. Prohibited for all other I/O Command Sets.

Redfish/Swordfish	NVMe / NVMe-oF
Notes	<p>Refer to the applicable NVMe I/O Command Set specification for details. For NVMe I/O Command Sets that don't define this field, it is considered reserved. For NVM Express Command Sets that Support Blocks: Bits 6:5 indicate the most significant 2 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace. If the NLBAF field is less than or equal to 16, then the host should ignore these bits. Bit 4 indicates whether or not the metadata is transferred at the end of the data LBA, creating an extended data LBA. (note: Bit 4 is not applicable when there is no metadata). Bits 3:0 indicate the least significant 4 bits of the Format Index of the supported LBA Format indicated in this data structure that was used to format the namespace.</p>

6.8.2.5 CapableSpeedGpbs

The mapping for `CapableSpeedGpbs` is summarized in Table 214.

Table 214: `CapableSpeedGpbs` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>CapableSpeedGpbs</code>	NVM Spec Property / Field: PCI Express Link Capabilities / PXCAP+Ch NVM Spec: Section:Figure NVMe PCIe Transport Specification 1.0b: 3.8.5.6: Figure 53: Bits 00:03
Type	Decimal	
Description	The speed, in gigabit per second (Gbit/s), at which this drive can communicate to a storage controller in ideal conditions.	indicates the supported Link speed(s) of the associated port.
LongDescription	This property shall contain fastest capable bus speed, in gigabit per second (Gbit/s), of the associated drive.	This field indicates the supported Link speed(s) of the associated port.
Mandatory	Mandatory	Mandatory
Notes		For NVMe-oF this is not specified; use the value for the highest supported native capability.

6.8.2.6 CapacityBytes

The mapping for `CapacityBytes` is summarized in Table 216.

For drives supporting only a single namespace (no or default endurance group / set only):

Table 215: CapacityBytes for single namespace mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CapacityBytes	NVM Spec Property / Field: – Identify Command / Identify Namespace Data Structure (CNS 00h), NVM Command Set / Namespace Size (NSZE) NVM Spec: Section:Figure NVMe 2.0: 5.17.1: Figure 273: CNS Value 00h NVM Command Set Specification 1.0b: 4.1.5.1: Figure 97: Bytes 00:07
Type	Int64	
Description	The size, in bytes, of this drive.	The total size of the NVM allocated its namespace.
LongDescription	This property shall contain the raw size, in bytes, of the associated drive.	The total size of the NVM allocated to this namespace in logical blocks. The value is in bytes. This field shall be supported if the Namespace Management capability is supported. This field may not correspond to the logical block size multiplied by the Namespace Size field. Due to thin provisioning or other settings (e.g., endurance), this field may be larger or smaller than the Namespace Size reported.
Mandatory	Mandatory	Optional
Notes	Reporting capacity in bytes is the Redfish and Swordfish standard mechanism.	

For drives supporting multiple namespaces:

Table 216: CapacityBytes for multiple namespace mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	CapacityBytes	NVM Spec Property / Field: Endurance Group Information Log (Log ID 09h) / Total Endurance Group Capacity (TEGCAP) NVM Spec: Section:Figure NVMe 2.0: 5.16.1.10: Figure 217: Bytes 160 : 175
Type	Int64	16 Bytes
Description	The number of bytes currently allocated by the storage system in this data store for this data type.	indicates the total NVM capacity in this Endurance Group.
LongDescription	The value shall be the number of bytes currently allocated by the storage system in this data store for this data type.	This field indicates the total NVM capacity in this Endurance Group.
Mandatory	Mandatory	No
Notes		This value is in bytes

6.8.2.7 Description The mapping for `Description` is summarized in Table 217.

Table 217: Description mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Description	N/A
Type	String	N/A
Description	The description of this resource.	N/A
LongDescription	This object represents the description of this resource. The resource values shall comply with the Redfish Specification-described requirements.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Description is a read-only field.	

6.8.2.8 EncryptionAbility

The mapping for `EncryptionAbility` is summarized in Table 218.

Table 218: EncryptionAbility mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	EncryptionAbility	N/A
Type	Enum (EncryptionAbility)	N/A
Description	The encryption ability of this drive.	N/A
LongDescription	This property shall contain the encryption ability for the associated drive.	N/A
Mandatory	Mandatory	
Notes	Required when encryption is supported. Available values: Non e/SelfEncryptingDrive/Other	Maps to vendor capabilities.

6.8.2.9 EncryptionStatus The mapping for `EncryptionStatus` is summarized in Table 219.

Table 219: EncryptionStatus mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	EncryptionStatus	N/A
Type	Drive.EncryptionStatus	N/A
Description	The status of the encryption of this drive.	N/A
LongDescription	This property shall contain the encryption status for the associated drive.	N/A
Mandatory	Optional Mandatory when EncryptionAbility != None	
Notes	Must be implemented and set when Encryption is enabled (EncryptionAbility will indicate encryption capability type.) Possible values: Unlocked /locked/foreign/unencrypted	Set according to vendor specs / mapping.

6.8.2.10 FailurePredicted

The mapping for `FailurePredicted` is summarized in Table 220.

Table 220: FailurePredicted mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	FailurePredicted	NVM Spec: Property / Field: – Persistent Event Log Log Identifier 0Dh) / NVM Subsystem Hardware Error Event (Event Type 05h) NVM Spec: Section :Figure NVMe 2.0: 5.16.1.14: Persistent Event Log (Log Identifier 0Dh) / Figure 224: Get Log Page – Persistent Event Log (Log Identifier 0Dh)NVMe 2.0: 5.16.1.14.1.5: Figure 232: NVM Subsystem Hardware Error Event Format: Bytes 00:01 (NVM Subsystem Hardware Error Event Code)
Type	Boolean	If any of the NVM Subsystem Hardware Events exist then this is a binary ‘true’
Description	An indication of whether this drive currently predicts a failure in the near future.	The Persistent Event Log page contains information about significant events not specific to a particular command.
LongDescription	This property shall indicate whether this drive currently predicts a manufacturer-defined failure.	The Persistent Event Log page contains information about significant events not specific to a particular command. The information in this log page shall be retained across power cycles and resets.
Mandatory	Recommended	Optional for Admin and I/O Controllers. Prohibited for Discovery Controllers
Notes	Implements the results of the SMART log data.	Reference NVMe 2.0: Section 5.16.1.14.1.5 Figure 233 for possible NVM Subsystem Hardware Error Event Codes

6.8.2.11 Identifiers The mapping for `Identifiers` is summarized in Table 221.

Table 221: Identifiers mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Identifiers	N/A
Type	Collection(Resource.Identifier)	N/A
Description	The Durable names for the subsystem.	N/A
LongDescription	This property shall contain a list of all known durable names for the associated subsystem.	N/A
Mandatory	Mandatory	
Notes	This is an array of unique identifiers for the NVM Subsystem.	This is an array of unique identifiers for the NVM Subsystem

6.8.2.12 Identifiers.DurableNameFormat The mapping for `Identifiers.DurableNameFormat` is summarized in Table 222.

Table 222: `Identifiers.DurableNameFormat` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Identifiers.DurableNameFormat</code>	N/A
Type	<code>Resource.v1_1_0.DurableNameFormat</code>	N/A
Description	The format of the Durable names for the subsystem.	N/A
LongDescription	This specifies the format of the associated NVM Subsystem NVMe Qualified Name of type NQN. Support for this field is mandatory if the controller supports revision 1.2.1 or later as indicated in the Version register (refer to section 3.1.2).	N/A
Mandatory	Optional	
Notes	This is an enum with multiple potential values. For this particular usage in Subsystem, there will only be one instance populated, of type NQN.	

6.8.2.13 Identifiers.DurableName

The mapping for `Identifiers.DurableName` is summarized in Table 223.

Table 223: `Identifiers.DurableName` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Identifiers.DurableName</code>	NVM Spec: Property / Field: NVMe 2.0: Identify Command / Identify Controller data structure (CNS 01h) / NVM Subsystem NVMe Qualified Name (SUBNQN) NVM Spec: Section :Figure NVMe 2.0: Section 5.17.2.1: Figure 275 - Bytes 768 - 1023)
Type	<code>Edm.String</code>	UTF-8 null-terminated string. Refer to NVMe 2.0: Section 4.5.1 Unique Identifier
Description	The format of the Durable names for the subsystem.	NVM Subsystem NVMe Qualified Name (SUBNQN)
LongDescription	This specifies the NVM Subsystem NVMe Qualified Name as a UTF-8 null-terminated string. Refer to NVMe Base Specification, section 7.9, for the definition of NVMe Qualified Name. Support for this field is mandatory if the controller supports revision 1.2.1 or later as indicated in the Version register (refer to section 3.1.2).	Used to uniquely describe a host or NVM subsystem for the purposes of identification and authentication.
Mandatory	Optional	Support for this field is mandatory if the controller supports revision 1.2.1 or later as indicated in the Controller Properties Version register (refer to section 3.1.3), Figure 35, Offset 8h.
Notes	For this particular usage in Subsystem, there will only be one instance populated in the <code>identifiers</code> array.	

6.8.2.14 IndicatorLED The mapping for `IndicatorLED` is summarized in Table 224.

Table 224: IndicatorLED mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	IndicatorLED	N/A
Type	Boolean	N/A
Description	An indication of whether this drive currently predicts a failure in the near future.	N/A
LongDescription	This property shall indicate whether this drive currently predicts a manufacturer-defined failure.	N/A
Mandatory	Do Not Implement	
Notes	This property has been deprecated. See guidance / implement the <code>LocationIndicatorActive</code> property instead.	

6.8.2.15 Links.Volumes The mapping for `Links.Volume` is summarized in Table 225.

Table 225: Links.Volume mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Links.Volume	N/A
Type	Collection(Volume.Volume)	N/A
Description	An array of links to the volumes that this drive either wholly or only partially contains.	N/A
LongDescription	This property shall contain an array of links to resources of type Volume with which this drive is associated. This property shall include all volume resources of which this drive is a member and all volumes for which this drive acts as a spare if the hot spare type is <code>Dedicated</code> .	N/A
Mandatory	Mandatory	
Notes	This array shall contain links to all namespaces associated with this physical drive.	

6.8.2.16 Links.Volumes@odata.count The mapping for `Links.Volumes@odata.count` is summarized in Table 226.

Table 226: `Links.Volumes@odata.count` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>Links.Volumes@odata.count</code>	N/A
Type	<code>(odata property)int64</code>	N/A
Description	Count of the number of items in the <code>Links.Volume</code> array.	N/A
LongDescription		N/A
Mandatory	Mandatory	
Notes		The number of namespaces is available from NVMe on a per controller basis.

6.8.2.17 Location The mapping for `Location` is summarized in Table 227.

Table 227: Location mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Location	N/A
Type	Collection(Resource.Location)	N/A
Description	The location of the drive.	N/A
LongDescription	This property shall contain location information of the associated drive.	N/A
Mandatory	Do Not Implement	
Notes	This property has been deprecated. See guidance / implement the PhysicalLocation property instead.	

6.8.2.18 LocationIndicatorActive The mapping for `LocationIndicatorActive` is summarized in Table 228.

Table 228: `LocationIndicatorActive` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>LocationIndicatorActive</code>	N/A
Type	Boolean	N/A
Description	An indicator allowing an operator to physically locate this resource.	N/A
LongDescription	This property shall contain the state of the indicator used to physically identify or locate this resource. A write to this property shall update the value of <code>IndicatorLED</code> in this resource, if supported, to reflect the implementation of the locating function.	N/A
Mandatory	Recommended	
Notes	This property replaces the <code>IndicatorLED</code> , which has been deprecated.	Comes from vendor.

6.8.2.19 Manufacturer The mapping for `Manufacturer` is summarized in Table 229.

Table 229: Manufacturer mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Manufacturer	NVM Spec Property / Field: Identify Command / Identify Controller Data structure (CNS 01h) / PCI Vendor ID (VID) NVM Spec: Section: Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 00:01
Type	String	16-bit number in little endian format.
Description	The manufacturer of this drive.	The company vendor identifier
LongDescription	This property shall contain the name of the organization responsible for producing the storage controller. This organization might be the entity from whom the storage controller is purchased, but this is not necessarily true.	The company vendor identifier that is assigned by the PCI SIG. This is the same value as reported in the ID register
Mandatory	Recommended	Mandatory
Notes		End clients expect to see the name of the company (e.g.; Contoso, BestVendor). While the value may be filled from the IdentifyController PCI Vendor ID or SubsystemID field, it would be preferable to have this filled with the actual string value of the company name.

6.8.2.20 MediaType The mapping for `MediaType` is summarized in Table 230.

Table 230: MediaType mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	MediaType	N/A
Type	enum (MediaType)	N/A
Description	The type of media contained in this Drive.	N/A
LongDescription	This property shall contain the type of media contained in the associated drive.	N/A
Mandatory	Mandatory	
Notes	Possible values: HDD/SSD/SMR.	NVMe SSD Drives to report SSD.

6.8.2.21 Metrics The mapping for `Metrics` is summarized in Table 231.

Table 231: Metrics mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Metrics	The DriveMetrics object contains multiple properties that map to NVMe properties, but the object itself does not have a direct mapping.
Type	DriveMetrics.DriveMetrics	
Description	The link to the metrics associated with this drive.	
LongDescription	This property shall contain a link to the metrics associated with this drive.	
Mandatory	Optional	
Notes		

6.8.2.22 Model The mapping for `Model` is summarized in Table 232.

Table 232: Model mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Model	NVM Spec Property / Field: Identify Command / Identify Controller Data Structure (CNS 01h) / Model Number (MN) NVM Spec: Section: Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Bytes 24:63
Type	String	String
Description	The model number for the Drive.	Model Number (MN)
LongDescription	This property shall contain the name by which the manufacturer generally refers to the Drive.	Contains the model number for the NVM subsystem that is assigned by the vendor as an ASCII string.
Mandatory	Mandatory	
Notes		Refer to NVMe Base specification 2.0 section 4.5.1 for unique identifier requirements. Refer to NVMe Base specification 2.0 section 1.4.2 for ASCII string requirements.

6.8.2.23 Multipath

The mapping for `Multipath` is summarized in Table 233.

Table 233: Multipath mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Multipath	<p>NVM Spec Property / Field: Identify Command / Identify Controller Data Structure (CNS 01h) / Controller Multi-Path I/O and Namespace Sharing Capabilities</p> <p>NVM Spec: Section:Figure NVMe 2.0: Section 5.17.2.1: Figure 275: Byte 76</p>
Type	Boolean	N/A
Description	An indication of whether the drive is accessible from multiple paths.	This field specifies multi-path I/O and namespace sharing capabilities of the controller and NVM subsystem
LongDescription	This property shall indicate whether the drive is accessible by an initiator from multiple paths allowing for failover capabilities upon a path failure.	<p>Bits 7:4 are reserved. Bit 3 if set to '1', then the NVM subsystem supports Asymmetric Namespace Access Reporting (refer to section NVMe Base Specification 2.0 section 8.1). If cleared to '0', then the NVM subsystem does not support Asymmetric Namespace Access Reporting. Bit 2 if set to '1', then the controller is associated with an SR-IOV Virtual Function. If cleared to '0', then the controller is associated with a PCI Function or a Fabrics connection. Bit 1 if set to '1', then the NVM subsystem may contain two or more controllers. If cleared to '0', then the NVM subsystem contains only a single controller. As described in refer to section NVMe Base Specification 2.0 section 2.4.1, an NVM subsystem that contains multiple controllers may be used by multiple hosts, or may provide multiple paths for a single host. Bit 0 if set to '1', then the NVM subsystem may contain more than one NVM subsystem port. If cleared to '0', then the NVM subsystem contains only a single NVM subsystem port.</p>

	Redfish/Swordfish	NVMe / NVMe-oF
Mandatory	Recommended Mandatory if drive is dual-ported.	Admin: Optional I/O: Optional Discovery: Reserved
Notes		

6.8.2.24 Name The mapping for `Name` is summarized in Table 234.

Table 234: Name mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Name	N/A
Type	String	N/A
Description	The name of the resource or array member.	N/A
LongDescription	This object represents the name of this resource or array member. The resource values shall comply with the Redfish Specification-described requirements. This string value shall be of the 'Name' reserved word format.	N/A
Mandatory	Mandatory	
Notes	In Redfish, Name is a read-only field.	

6.8.2.25 NegotiatedSpeedGbps

The mapping for `NegotiatedSpeedGbps` is summarized in Table 235.

Table 235: NegotiatedSpeedGbps mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	NegotiatedSpeedGbps	For NVMe: NVM Spec Property / Field: : For PCIe Transport: Offset PXCAP + Ch: PXLCAP – PCI Express Link Capabilities / Supported Link Speeds (SLS) NVM Spec: Section:Figure For PCIe Transport: NVM Express PCIe Transport Specification 1.0b: section 3.8.5.6: Figure 53: Bytes 03:00For NVMe-oF: Not Applicable
Type	Decimal	
Description	The speed, in gigabit per second (Gbit/s), at which this drive currently communicates to the storage controller.	This field indicates the supported Link speed(s) of the associated port
LongDescription	This property shall contain current bus speed, in gigabit per second (Gbit/s), of the associated drive.	Supported Link Speeds
Mandatory	Mandatory	For NVMe: Required For NVMe-oF: DNI
Notes		For PCIe, this is in the PCIe link capabilities For NVMe-oF this is not specified; use the value for the native capability.

6.8.2.26 PhysicalLocation.Info The mapping for `PhysicalLocation.Info` is summarized in Table 236.

Table 236: `PhysicalLocation.Info` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PhysicalLocation.Info</code>	N/A
Type	String	N/A
Mandatory	Do Not Implement	
Notes	This property has been deprecated.	

6.8.2.27 PhysicalLocation.InfoFormat The mapping for `PhysicalLocation.InfoFormat` is summarized in Table 237.

Table 237: `PhysicalLocation.InfoFormat` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PhysicalLocation.InfoFormat</code>	N/A
Type	String	N/A
Mandatory	Do Not Implement	
Notes	This property has been deprecated.	

6.8.2.28 PhysicalLocation.PartLocation The mapping for `PhysicalLocation.PartLocation` is summarized in Table 238.

Table 238: `PhysicalLocation.PartLocation` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PhysicalLocation.PartLocation</code>	N/A
Type	Boolean	N/A
Description	An indication of whether the drive is accessible from multiple paths.	N/A
LongDescription	This property shall indicate whether the drive is accessible by an initiator from multiple paths allowing for failover capabilities upon a path failure.	N/A
Mandatory	Recommended Mandatory if drive is dual-ported.	
Notes	The drive should support this property to be filled in by a layered process (e.g., BMC). Support for any other properties in <code>PhysicalLocation</code> are at the discretion of the vendor.	

6.8.2.29 PredictedMediaLifetimeLeftPercent The mapping for `PredictedMediaLifetimeLeftPercent` is summarized in Table 239.

Table 239: `PredictedMediaLifetimeLeftPercent` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>PredictedMediaLifetimeLeftPercent</code>	NVM Spec Property / Field: Get Log Page – SMART, Health Information Log (Log Identifier 02h) / Percentage Used NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3: Figure 207: Byte 05
Type	Decimal	Decimal range: 0% - 255%
Description	The percentage of reads and writes that are predicted to be available for the media.	Contains a vendor specific estimate of the percentage of NVM subsystem life used based on the actual usage and the manufacturer’s prediction of NVM life.
LongDescription	This property shall contain an indicator of the percentage of life remaining in the drive’s media.	Contains a vendor specific estimate of the percentage of NVM subsystem life used based on the actual usage and the manufacturer’s prediction of NVM life. A value of 100 indicates that the estimated endurance of the NVM in the NVM subsystem has been consumed, but may not indicate an NVM subsystem failure. The value is allowed to exceed 100. Percentages greater than 254 shall be represented as 255. This value shall be updated once per power-on hour (when the controller is not in a sleep state).
Mandatory	Mandatory	Optional
Notes	Maps to percentage used in SMART information log	This value shall be scaled to a value between 0% - 100% Calculate resulting value as 100 - value reported (PercentageUsed). Refer to the JEDEC JESD218A standard for SSD device life and endurance measurement techniques.

6.8.2.30 Protocol The mapping for `Protocol` is summarized in Table 240.

Table 240: Protocol mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Protocol	N/A
Type	Protocol.Protocol	N/A
Description	The protocol that this drive currently uses to communicate to the storage controller.	N/A
LongDescription	This property shall contain the protocol that the associated drive currently uses to communicate to the storage controller for this system.	N/A
Mandatory	Mandatory	
Notes	Possible values (long list)	NVMe Drives shall report “NVMe”.

6.8.2.31 Revision The mapping for `Revision` is summarized in Table 241.

Table 241: Revision mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Revision	NVM Spec Property / Field: Identify Command / Identify Controller Data Structure (CNS 01h) / Firmware Revision (FR) NVM Spec: Section:Figure NVMe 2.0: section 5.17.2.1: Figure 275: Bytes 64:71
Type	String	String
Description	The revision of this drive. This is typically the firmware or hardware version of the drive.	Contains the currently active firmware revision, as an ASCII string, for the domain of which this controller is a part.
LongDescription	This property shall contain the manufacturer-defined revision for the associated drive.	Contains the currently active firmware revision, as an ASCII string, for the domain of which this controller is a part.
Mandatory	Mandatory	Mandatory
Notes		Return the currently active firmware revision information. This is the same revision information that may be retrieved with the Get Log Page command, refer to NVMe Base specification 2.0, section 5.16.1.4

6.8.2.32 RotationSpeedRPM The mapping for `RotationSpeedRPM` is summarized in Table 242.

Table 242: `RotationSpeedRPM` mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	<code>RotationSpeedRPM</code>	NVM Spec Property / Field: Get Log Page - Rotational Media Information Log (Log ID 16h) / Nominal Rotational Speed (NRS) NVM Spec: Section:Figure NVMe 2.0: section 5.16.1.22: Figure 263: Bytes 4:5
Type	Decimal	Decimal
Description	An indication of whether the drive is accessible from multiple paths.	Nominal rotational speed in revolutions per minute while the current Power State is 0
LongDescription	This property shall indicate whether the drive is accessible by an initiator from multiple paths allowing for failover capabilities upon a path failure.	This log page provides rotational media information (refer to section 8.20) for Endurance Groups that store data on rotational media. The information provided is retained across power cycles and resets.
Mandatory	Optional If MediaType == SSD, Do Not Implement.	I/O Controller: Optional Admin Controller: Prohibited Discovery Controller: Prohibited
Notes		Nominal rotational speed in revolutions per minute while the current Power State is 0 Supported Values / Descriptions: 0000h Not reported 0001h This value shall not be used. FFFFh Reserved All other values Nominal rotational speed in revolutions per minute while the current Power State is 0

6.8.2.33 SKU The mapping for `SKU` is summarized in Table 243.

Table 243: SKU mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SKU	N/A
Type	String	N/A
Description	The SKU for this drive.	N/A
LongDescription	This property shall contain the stock-keeping unit (SKU) number for this drive.	N/A
Mandatory	Mandatory	
Notes	The drive should support this property to be filled in by a layered process (e.g., OEM manufacturing).	

6.8.2.34 SerialNumber

The mapping for `SerialNumber` is summarized in Table 244.

Table 244: SerialNumber mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SerialNumber	NVM Spec Property / Field: Identify Command / Identify Controller Data Structure (CNS 01h): Serial Number (SN) NVM Spec: Section:Figure NVMe 2.0: section 5.17.2.1: Figure 275: Bytes 04:23
Type	String	String
Description	The serial number for this drive.	Contains the serial number for the NVM subsystem that is assigned by the vendor as an ASCII string.
LongDescription	This property shall contain the manufacturer-allocated number that identifies the drive.	Contains the serial number for the NVM subsystem that is assigned by the vendor as an ASCII string.
Mandatory	Mandatory	I/O Controller: Mandatory Admin Controller: MandatoryDiscovery Controller: Restricted
Notes		Each identifier is in big endian format. Refer to NVMe 2.0 Base specification section 4.5.1 for unique identifier requirements. Refer to NVMe 2.0 Base specification section 1.4.2 for ASCII string requirements

6.8.2.35 Status.State

The mapping for `Status.state` is summarized in Table 245.

Table 245: Status.State mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.State	NVM Spec Property / Field: Property Get Command / CSTS (Offset 1Ch) – Controller Status: Ready (RDY) - NVM Spec: Section:Figure NVMe 2.0: section 3.1.3.6: Figure 47: Bit 00
Type	Resource.State (enum)	Enum
Description	The known state of the resource, such as, enabled.	This bit is set to ‘1’ when the controller is ready to process submission queue entries after CC.EN is set to ‘1’. This bit shall be cleared to ‘0’ when CC.EN is cleared to ‘0’ once the controller is ready to be re-enabled
LongDescription	This property shall indicate whether and why this component is available. Enabled indicates the resource is available. Disabled indicates the resource has been intentionally made unavailable but can be enabled. Offline indicates the resource is unavailable intentionally and requires action to make it available. InTest indicates that the component is undergoing testing. Starting indicates that the resource is becoming available. Absent indicates the resource is physically unavailable.	This bit is set to ‘1’ when the controller is ready to process submission queue entries after CC.EN is set to ‘1’. This bit shall be cleared to ‘0’ when CC.EN is cleared to ‘0’ once the controller is ready to be re-enabled
Mandatory	Optional Mandatory for NVM Drives	Mandatory
Notes	Possible values: Enabled / Disabled / StandbyOffline / StandbySpare / InTest / Starting / Absent / UnavailableOffline / Deferring / Quiesced / Updating / Qualified	The drive should support this property to be filled in by a higher level client (e.g., BMC). The drive can self-set this drive to Enabled / Disabled / InTest / Updating. If any controller in the drive is set to Enabled, set to “Enabled”. If all controllers are set to disabled, set to “Disabled”. If a firmware update is in progress, set to “Updating”. If the drive is running a self-test, set to “InTest”.

6.8.2.36 Status.Health

The mapping for `Status.Health` is summarized in Table 246.

Table 246: Status.Health mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	Status.Health	NVM Spec Property / Field: Get Log Page – SMART / Health Information Log (Log ID 02h) / Critical Warning NVM Spec: Section:Figure NVMe 2.0: Section 5.16.1.3, Figure 207: Byte 00
Type	Resource.Health	Byte field
Description	The health state of this resource in the absence of its dependent resources.	This field indicates critical warnings for the state of the controller. Each bit corresponds to a critical warning type
LongDescription	This property shall represent the health state of the resource without considering its dependent resources. The values shall conform to those defined in the Redfish Specification.	This field indicates critical warnings for the state of the controller. Each bit corresponds to a critical warning type; multiple bits may be set to '1'. If a bit is cleared to '0', then that critical warning does not apply. Critical warnings may result in an asynchronous event notification to the host. Bits in this field represent the current associated state and are not persistent.
Mandatory	Optional Mandatory for NVM Drives.	I/O Controller: MandatoryAdmin Controller: OptionalDiscovery Controller: Prohibited

	Redfish/Swordfish	NVMe / NVMe-oF
Notes	Possible Values: OK / Warning / Critical	Report to same value as set for worst-case controller Status.Health. Bits / Definitions7:6 Reserved5 If set to '1', then the Persistent Memory Region has become read-only or unreliable.4 If set to '1', then the volatile memory backup device has failed. This field is only valid if the controller has a volatile memory backup solution.3 If set to '1', then all of the media has been placed in read only mode. The controller shall not set this bit to '1' if the read-only condition on the media is a result of a change in the write protection state of a namespace.2 If set to '1', then the NVM subsystem reliability has been degraded due to significant media related errors or any internal error that degrades NVM subsystem reliability.1 If set to '1', then a temperature is: a) greater than or equal to an over temperature threshold; or b) less than or equal to an under temperature threshold.0 If set to '1', then the available spare capacity has fallen below the threshold.

6.8.2.37 StatusIndicator The mapping for `StatusIndicator` is summarized in Table 247.

Table 247: StatusIndicator mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	StatusIndicator	N/A
Type	enum (StatusIndicator)	N/A
Description	An indication of whether the drive is accessible from multiple paths.	N/A
LongDescription	This property shall indicate whether the drive is accessible by an initiator from multiple paths allowing for failover capabilities upon a path failure.	N/A
Mandatory	Mandatory	
Notes	Multiple values possible. Relevant values for NVMe standalone drives: OK, Fail, PredictiveFailureAnalysis.	The drive should support this property to be filled in by a higher level client (e.g., BMC). The drive can self-set this drive only to OK (or potentially fail, but only if the property has not been set by higher-level software).

6.8.2.38 WriteCacheEnabled

The mapping for WriteCacheEnabled is summarized in Table 248.

Table 248: WriteCacheEnabled mapping

	Redfish/Swordfish	NVMe / NVMe-oF
Property	SerialNumber	NVM Spec Property / Field: Get Features / Volatile Write Cache (Feature Identifier 06h) / Volatile Write Cache Enable (WCE) NVM Spec: Section: Figure NVMe 2.0: Section 5.27.1.4: Figure 321: Bit 00
Type	Boolean	Boolean
Description	An indication of whether the drive write cache is enabled.	This property shall indicate whether the drive write cache is enabled.
LongDescription	This property shall indicate whether the drive write cache is enabled.	This property shall indicate whether the drive write cache is enabled.
Mandatory	Optional Mandatory for NVMe Drives.	Optional
Notes		If set to '1', then the volatile write cache is enabled. If cleared to '0', then the volatile write cache is disabled

7 Other Feature Mapping

7.1 Introduction

There are additional features and functionality that are also desirable to be exposed via the Redfish/Swordfish management ecosystems; however, there is not necessarily a clear mapping for these to a specific portion of the NVMe specification, or the functionality has a large component provided by the NVMe device implementer. This section provides an overview of the required functionality and presentation required to present a common, standardized interface to the Redfish/Swordfish clients.

7.2 Firmware Update

Firmware update is a required function to present comprehensive management. This is presented in Redfish through the “Update” service. The Update Service provides a standardized interface to all software and firmware update mechanisms throughout the server, storage, and networking environment, and as such, provides an abstracted, standardized interface that largely manages the acquisition from the client of the image files (and subsequent distribution of those to the designated targets), as well as very coarse-grained scheduling.

Note that this mechanism does not include any “active image” management capability (e.g., rollback).

7.2.1 Firmware update for NVMe Drives

NVMe drives shall support a standard implementation of the Redfish Update service. This is fundamentally a single-step invocation process on the update service that gets the image file from a specified URI using the SimpleUpdate action.

One parameter that shall be specified is the OperationApplyTime; this indicates when to apply the image. If not explicitly specified, the implementation should assume the request is to be applied immediately.

The values of the OperationApplyTime that shall be supported by NVMe drives are: - Immediate - OnReset

Example simple update request:

```
POST /redfish/v1/UpdateService/Actions/UpdateService.SimpleUpdate HTTP/1.1
Content-Type: application/json
Content-Length: <computed-length>

{
  "ImageURI": "https://192.168.1.250/images/dev1_update.bin"
  "@Redfish.OperationApplyTime": "OnReset"
}
```

The `username` and `password` parameters summarized in Table 249 may be required to access the image. Implementations should support these. Additionally, the implementation may support the `Targets` parameter to specify the list of devices on which to apply the image. If `Targets` is not specified, the image will be applied to all applicable devices.

Table 249: Additional parameters

Parameter	Type	Description
Username	String	The user name to access the URI specified by the <code>ImageURI</code> parameter.
Password	String	The password to access the URI specified by the <code>ImageURI</code> parameter.
Targets	Array	An array of strings that are URIs to resources that indicate where to apply the image.

The SimpleUpdate mechanism is fundamentally a “pull” mechanism (the implementation pulls a file from a target); the UpdateService also supports a “push” mechanism (the client “pushes” the image to the implementation as part of the request payload) that may also be implemented. For more information this mechanism, using the `MultipartHttpPushUri` property, is documented in the Redfish Specification.

7.2.1.1 Mapping Images to NVMe Capabilities It will be up to the implementation to determine the NVMe specific parameters to use upon receipt of the image from the update service. However, the following general guidelines should be used to ensure behavioral consistency across implementations:

When receiving image: - If open slots, the implementation can choose what open slot to use, and load the image to that slot. - If all slots are full, the implementation can choose (typically any slot except the currently active image).

After loaded to a slot, the implementation will mark the image as active. Based on the “apply time” setting, the implementation setting will activate the image either immediately, or set to activate on (the next) reset.

7.3 Security Management Protocols

NVMe specifies an optional channel for security protocol communication with controller devices, implemented as a pair of send and receive functions: Security Send (described in section 5.26 of NVMe) and Security Receive (section 5.25). These provide functionality equivalent to the SPC-5 `SECURITY_PROTOCOL_IN` and `SECURITY_PROTOCOL_OUT` commands.

Swordfish maps these Security Send and Security Receive commands to two StorageController actions:

- `#StorageController.SecuritySend`
- `#StorageController.SecurityReceive`

Swordfish restricts the allowable security protocol numbers to 0, 1 or 2 for the Security Receive action, and 1 or 2 for the Security Send action.

Where the security protocol data is passed as a base64-encoded string, opaque to the Swordfish/Redfish implementation.

Appendix A: Objects without a direct mapping to the NVMe model

A.1: Overview

There are a number of objects that are required for the proper integration of NVMe support within Redfish and Swordfish, but which are not supported by an entity that can be mapped directly from the various NVM Specifications. Information about these related but un-mapped objects are defined elsewhere in the documentation provided with each Swordfish release.

They are summarized in Table A.1 for ease of use, including:

- Object Name
- NVMe Device Expected Usage

Object	NVMe Device Expected Usage
NetworkAdapter	NVMe-oF Subsystems, Network-Attach Drives
Port (on NetworkAdapter)	NVMe-oF Subsystems, Network-Attach Drives
NetworkDeviceFunction	NVMe-oF Subsystems, Network-Attach Drives
Fabric	NVMe-oF Subsystems, Network-Attach Drives and arrays
Connection	NVMe-oF Subsystems, Network-Attach Drives and arrays
Endpoint	NVMe-oF Subsystems, Network-Attach Drives and arrays
Endpoint Group	NVMe-oF Subsystems, Network-Attach Drives and arrays
Switch	NVMe-oF Subsystems, Network-Attach Drives and arrays
Port (on switch)	NVMe-oF Subsystems, Network-Attach Drives and arrays
EthernetInterface	NVMe-oF Subsystems, Network-Attach Drives and arrays, EBOF/JBOF
Manager	EBOF/JBOF, Complex devices, Arrays
NetworkProtocol	EBOF/JBOF, Complex devices, Arrays

Table A.1: Related Objects

Required usage of objects and their properties are contained in the relevant Swordfish profiles.

For example, for Ethernet-Attach drives, the NetworkAdapter, Port, and NetworkDeviceFunction required properties are specified as part of the “Ethernet-Attach NVMe Drive” profile.

A.2: Related Use Cases

The Swordfish Scalable Storage Management API User’s Guide contains real-world examples of how clients will interact with an implementation. Table A.2 summarizes the NVMe use cases as of 20 June 2023.

Title	Description
Attach a Namespace	Attach a Namespace
Deprovision a Namespace	Deprovision a Namespace
Detach a Namespace	Detach a Namespace
Provision a Namespace	Provision a Namespace
Provision an NVM Set with a Namespace	Provision an NVM Set associated with a Namespace
Report capacity for a Namespace	Report capacity for a Namespace
Report remaining life for a Namespace	Report remaining life for a Namespace

Table A.2: NVMe Use Case Summary

Annex B: Bibliography

B.1 Overview

The following referenced documents provide important support for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

B.2 Informational references

The informational references are summarized in Table B.1.

Tag	Title (Version)	Author	URL
Pro files	Swordfish Profile Bundle Working Draft	SNIA	< https://www.snia.org/forums/smi/swordfish >
Users Guide	Swordfish Scalable Storage Management API User's Guide	SNIA	< https://www.snia.org/forums/smi/swordfish >

Table B.1: Informational References