



Storage Management Technical Specification, Part 3 Common Profiles

Version 1.7.0, Revision 5

Abstract: This SNIA Technical Position defines an interface between WBEM-capable clients and servers for the secure, extensible, and interoperable management of networked storage.

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. Suggestions for revision should be directed to <http://www.snia.org/feedback/>.

SNIA Technical Position

8 March, 2016

REVISION HISTORY

Revision 1

Date

8 Sept 2014

SCRs Incorporated and other changes

Experimental Indication (CORE-SMIS-SCR-00081)

- Deprecated this profile

Indications Profile (SMIS-170-Draft-SCR00002)

- Cleaned up the SNIA specialization of the DMTF Indications profile--removed unused Experimental content (SNIA_IndicationFilterTemplates, Properties added to AlertIndication, Properties to CIM_IndicationSubscription, CIM_HostedCollection for FilterCollections, Pre-defined FilterCollections, "Client defined" FilterCollections, SNIA_IndicationConfigurationService, SNIA_IndicationConfigurationCapabilities and Indication correlation).

Object Manager Adapter (CORE-SMIS-SCR-00081)

- Deprecated this profile

References (CORE-SMIS-SCR-00081)

- Added a reference for the DMTF WBEM Server Profile
- Updated some references that moved from "under development" to "final"

Server (CORE-SMIS-SCR-00081)

- Deprecated this profile

WBEM Server (CORE-SMIS-SCR-00081)

- Added this profile and noted that it is required for all SMI-S implementations

SMIS-160-Errata-SCR00010

Comments

Editorial notes and DRAFT material are displayed.

Revision 2

Date

18 December 2014

SCRs Incorporated and other changes

iSCSI Target Ports Subprofile (SMIS-161-Errata-SCR00002)

- Fixed typo: For *CIM_ElementCapabilities*, *USAGE=CIM_iSCSICapabilities to System*

Comments

Editorial notes and DRAFT material are displayed.

USAGE text was revised to address code.
(now included in the front matter for all SNIA specifications)

Revision 3

Date

20 May 2015

SCRs Incorporated and other changes

All recipes and associated text were removed.

TSG-SMIS-SCR00315.001

- Promoted the maturity level from DRAFT to EXPERIMENTAL for these revisions--Update profiles to remove SNIA_ classes and use DMTF CIM_ classes in these profiles: Common, SAS Target Port, SB Target Ports, FC-SB-x Initiator Ports, Generic Initiator Ports, iSCSI Initiator Ports, FC Initiator Ports, SAS Initiator Ports, ATA Initiator Ports, SB Initiator Ports, FCoE Initiator Ports, Cascading, Server Profile, Experimental Indications, Proxy Server System Management, Operational Power, Indications.

Base Server Profile

- Changed references from *DSP1011 version 1.0.1* to *DSP1011 version 1.0.2*

Cascading Subprofile

- Deleted the text of this profile (was deprecated) and added a reference to the last (non-deprecated) version of this profile, in SMI-S Version 1.4.0 Revision 6.

FC Target Ports Profile

- SMI Referenced Properties/Methods for CIM_FCPort Table, Row for NetworkAddresses (added): text changed to "8 un-separated upper case hex digits"

Proxy Server System Management Subprofile

- Added a reference to the last version of this profile in SMI-S Version 1.6.1 Revision 5.

References

- Added *DMTF DSP1054 v1.2.2, Indications Profile* (and changed version to 1.2.2 throughout book)

Annex A (informative) SMI-S Information Mode

- Removed statement about SNIA_ classes.
- Changed reference to CIM v2.44

MultiSystem Subprofile

- SMI Referenced Properties/Methods for CIM_RedundancySet table: RedundancyStatus row revised. (SMIS-160-Errata-SCR00010)

Comments

Editorial notes and DRAFT material were hidden.

Revision 4

Date

9 September 2015

SCRs Incorporated and other changes

Multiple profiles

- Instances of *subprofile* were changed to *profile*. In the annex, instances of *subprofile* were changed to *component profile*. (TSG meeting voice vote)
- Profile versions and related text were updated. (TSG meeting voice vote)
- *CIM/XML* was changed to *CIM-XML* (Response to ballot comments)

ATA Initiator Ports

- Deprecated this experimental profile (TSG-SMIS-SCR00318)

Backend Ports

- Removed this obsolete and deprecated profile (TSG-SMIS-SCR00318)

Cluster Profile

- Removed this obsolete and deprecated profile (TSG-SMIS-SCR00318)

Direct Attach (DA) Ports Profile

- Promoted to Stable (TSG-SMIS-SCR00318)

Extra Capacity Set

- Removed this obsolete and deprecated profile (TSG-SMIS-SCR00318)

Fan Profile

FC Initiator Ports

- Promoted to Stable (TSG-SMIS-SCR00318)

FCoE Target Ports Profile

- Moved within Part 3 to follow FC Target Ports Profile (CORE-SMIS-SCR-00084)
- Text tweaked for FCoE, including test and figure referencing ProtocolControllerForPort (CORE-SMIS-SCR-00084)

Generic Initiator Ports

- Promoted to Stable (TSG-SMIS-SCR00318)

Health Package

- Promoted sections 25.1.8 (RECE) and 25.1.6 (TSG-SMIS-SCR00318)

Indication Profile

- Removed this obsolete and deprecated profile (TSG-SMIS-SCR00318)

iSCSI Initiator Port Profile

- Promoted to Stable (TSG-SMIS-SCR00318)

Miscellaneous Security Profiles

- Removed this obsolete and deprecated profile (TSG-SMIS-SCR00318)

Parallel SCSI (SPI) Target Ports Profile

- Removed per von Behren SMI-S V1.7r3 comments 2 and 3

Power Supply Profile

- Promoted to Stable (TSG-SMIS-SCR00318)

Recipe Overview

- Removed this obsolete and deprecated profile (TSG-SMIS-SCR00318)

References

- Updated some references here and within profiles, removed others (irrelevant)

SAS Initiator Ports Profile

- Promoted to Stable (TSG-SMIS-SCR00318)

Serial Attached SCSI (SAS) Target Port Profile (LSI/NetApp and HP implementations)

- Promoted to Stable (TSG-SMIS-SCR00318)

SATA Target Port Profile

- Deprecated this experimental profile (TSG-SMIS-SCR00318)

SPI Initiator Ports Profile

- Deprecated this experimental profile (TSG-SMIS-SCR00318)

SPI Target Ports Profile

- Deprecated this experimental profile (TSG-SMIS-SCR00318)
- Promoted to Stable (TSG-SMIS-SCR00318)

Annex A (informative) SMI-S Information Model

- DMTF's CIM schema version changed to 2.45.0. (TSG meeting voice vote)

Annex B Cross Profile Considerations

- Added: content moved within Part 3 to Annex B. (CORE-SMIS-SCR-00084)

Comments

- Editorial notes and DRAFT material were hidden.

Revision 5

Date

9 October 2015

SCRs Incorporated and other changes

Multiple profiles: Addressed SMI-S 1.7.0 Revision 4 TSG ballot comments that were strictly editorial and were approved by voice vote of the TSG.

References

- Removed DSP0004, DSP0200, DSP0202, DSP0207

Comments

- Editorial notes were hidden.

Suggestion for changes or modifications to this document should be sent to the SNIA Storage Management Initiative Technical Steering Group (SMI-TSG) at <http://www.snia.org/feedback/>

USAGE

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 shall 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 shall acknowledge the SNIA copyright on that material, and shall 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, sell any or this entire document, 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 e-mailing tcmd@snia.org. Please include the identity of the requesting individual and/or company and a brief description of the purpose, nature, and scope of the requested use.

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

BSD 3-Clause Software License

Copyright (c) 2016, The Storage Networking Industry Association.

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

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of The Storage Networking Industry Association (SNIA) nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

DISCLAIMER

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

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

Copyright © 2003-2016 SNIA. All rights reserved. All other trademarks or registered trademarks are the property of their respective owners.

Portions of the CIM Schema are used in this document with the permission of the Distributed Management Task Force (DMTF). The CIM classes that are documented have been developed and reviewed by both the SNIA and DMTF Technical Working Groups. However, the schema is still in development and review in the DMTF Working Groups and Technical Committee, and subject to change.

INTENDED AUDIENCE

This document is intended for use by individuals and companies engaged in developing, deploying, and promoting interoperable multi-vendor SANs through the Storage Networking Industry Association (SNIA) organization.

CHANGES TO THE SPECIFICATION

Each publication of this specification is uniquely identified by a three-level identifier, comprised of a version number, a release number and an update number. The current identifier for this specification is version 1.7.0. Future publications of this specification are subject to specific constraints on the scope of change that is permissible from one publication to the next and the degree of interoperability and backward compatibility that should be assumed between products designed to different publications of this standard. The SNIA has defined three levels of change to a specification:

- **Major Revision:** A major revision of the specification represents a substantial change to the underlying scope or architecture of the SMI-S API. A major revision results in an increase in the version number of the version identifier (e.g., from version 1.x.x to version 2.x.x). There is no assurance of interoperability or backward compatibility between releases with different version numbers.
- **Minor Revision:** A minor revision of the specification represents a technical change to existing content or an adjustment to the scope of the SMI-S API. A minor revision results in an increase in the release number of the specification's identifier (e.g., from x.1.x to x.2.x). Minor revisions with the same version number preserve interoperability and backward compatibility.
- **Update:** An update to the specification is limited to minor corrections or clarifications of existing specification content. An update will result in an increase in the third component of the release identifier (e.g., from x.x.1 to x.x.2). Updates with the same version and minor release levels preserve interoperability and backward compatibility.

TYPOGRAPHICAL CONVENTIONS

Maturity Level

In addition to informative and normative content, this specification includes guidance about the maturity of emerging material that has completed a rigorous design review but has limited implementation in commercial products. This material is clearly delineated as described in the following sections. The typographical convention is intended to provide a sense of the maturity of the affected material, without altering its normative content. By recognizing the relative maturity of different sections of the standard, an implementer should be able to make more informed decisions about the adoption and deployment of different portions of the standard in a commercial product.

This specification has been structured to convey both the formal requirements and assumptions of the SMI-S API and its emerging implementation and deployment lifecycle. Over time, the intent is that all content in the specification will represent a mature and stable design, be verified by extensive implementation experience, assure consistent support for backward compatibility, and rely solely on content material that has reached a similar level of maturity. Unless explicitly labeled with one of the subordinate maturity levels defined for this specification, content is assumed to satisfy these requirements and is referred to as "Finalized". Since much of the evolving specification

content in any given release will not have matured to that level, this specification defines three subordinate levels of implementation maturity that identify important aspects of the content's increasing maturity and stability. Each subordinate maturity level is defined by its level of implementation experience, its stability and its reliance on other emerging standards. Each subordinate maturity level is identified by a unique typographical tagging convention that clearly distinguishes content at one maturity model from content at another level.

Experimental Maturity Level

No material is included in this specification unless its initial architecture has been completed and reviewed. Some content included in this specification has complete and reviewed design, but lacks implementation experience and the maturity gained through implementation experience. This content is included in order to gain wider review and to gain implementation experience. This material is referred to as “Experimental”. It is presented here as an aid to implementers who are interested in likely future developments within the SMI specification. The contents of an Experimental profile may change as implementation experience is gained. There is a high likelihood that the changed content will be included in an upcoming revision of the specification. Experimental material can advance to a higher maturity level as soon as implementations are available. Figure 1 is a sample of the typographical convention for Experimental content.

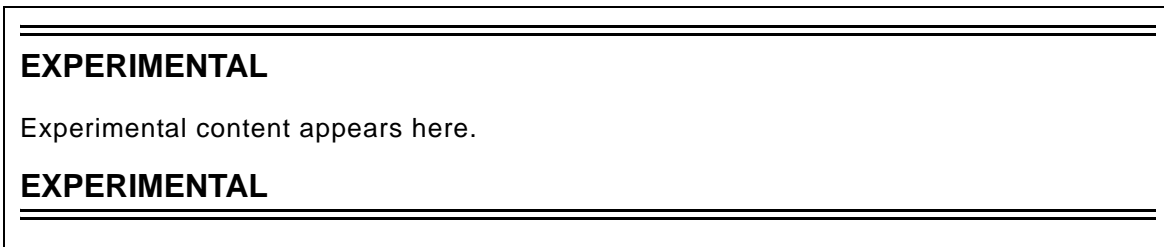


Figure 1 - Experimental Maturity Level Tag

Implemented Maturity Level

Profiles for which initial implementations have been completed are classified as “Implemented”. This indicates that at least two different vendors have implemented the profile, including at least one provider implementation. At this maturity level, the underlying architecture and modeling are stable, and changes in future revisions will be limited to the correction of deficiencies identified through additional implementation experience. Should the material become obsolete in the future, it must be deprecated in a minor revision of the specification prior to its removal from subsequent releases. Figure 2 is a sample of the typographical convention for Implemented content.

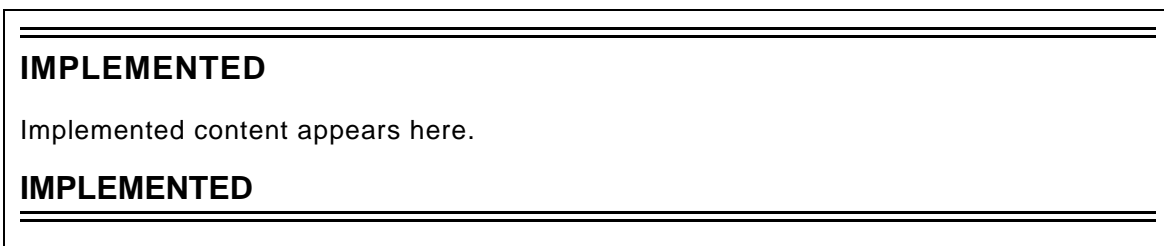


Figure 2 - Implemented Maturity Level Tag

Stable Maturity Level

Once content at the Implemented maturity level has garnered additional implementation experience, it can be tagged at the Stable maturity level. Material at this maturity level has been implemented by three different vendors, including both a provider and a client. Should material that has reached this maturity level become obsolete, it may only be deprecated as part of a minor revision to the specification. Material at this maturity level that has been deprecated may only be removed from the specification as part of a major revision. A profile that has reached this maturity level is guaranteed to preserve backward compatibility from one minor specification revision to the next. As a result, Profiles at or above the Stable

maturity level shall not rely on any content that is Experimental. Figure 3 is a sample of the typographical convention for Implemented content.



Figure 3 - Stable Maturity Level Tag

Finalized Maturity Level

Content that has reached the highest maturity level is referred to as “Finalized.” In addition to satisfying the requirements for the Stable maturity level, content at the Finalized maturity level must solely depend upon or refine material that has also reached the Finalized level. If specification content depends upon material that is not under the control of the SNIA, and therefore not subject to its maturity level definitions, then the external content is evaluated by the SNIA to assure that it has achieved a comparable level of completion, stability, and implementation experience. Should material that has reached this maturity level become obsolete, it may only be deprecated as part of a major revision to the specification. A profile that has reached this maturity level is guaranteed to preserve backward compatibility from one minor specification revision to the next. Over time, it is hoped that all specification content will attain this maturity level. Accordingly, there is no special typographical convention, as there is with the other, subordinate maturity levels. Unless content in the specification is marked with one of the typographical conventions defined for the subordinate maturity levels, it should be assumed to have reached the Finalized maturity level.

Deprecated Material

Non-Experimental material can be deprecated in a subsequent revision of the specification. Sections identified as “Deprecated” contain material that is obsolete and not recommended for use in new development efforts. Existing and new implementations may still use this material, but shall move to the newer approach as soon as possible. The maturity level of the material being deprecated determines how long it will continue to appear in the specification. Implemented content shall be retained at least until the next revision of the specialization, while Stable and Finalized material shall be retained until the next major revision of the specification. Providers shall implement the deprecated elements as long as it appears in the specification in order to achieve backward compatibility. Clients may rely on deprecated elements, but are encouraged to use non-deprecated alternatives when possible.

Deprecated sections are documented with a reference to the last published version to include the deprecated section as normative material and to the section in the current specification with the replacement. Figure 4 contains a sample of the typographical convention for deprecated content.

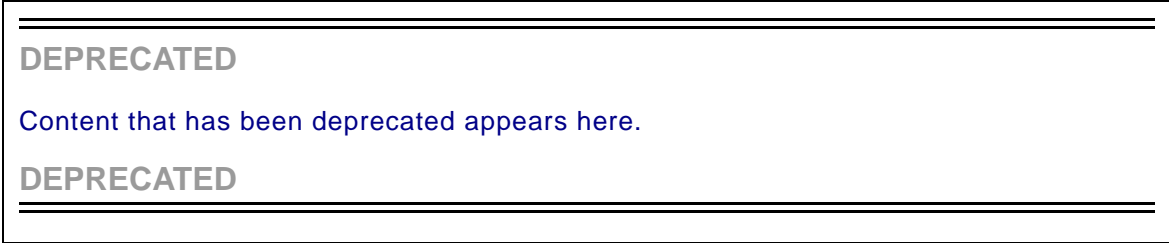


Figure 4 - Deprecated Tag

Contents

Revision History	2
List of Figures	19
List of Tables	21
Foreword	31
1 Scope	33
2 Normative References	35
2.1 Approved References	35
2.2 DMTF References (Final).....	35
2.3 References under development	35
3 Terms and Definitions.....	37
3.1 General	37
3.2 Terms	37
4 Profile Introduction.....	39
4.1 Profile Overview	39
4.2 Format for Profile Specifications	40
5 Generic Target Ports Profile	43
5.1 Synopsis.....	43
5.2 Description	43
5.3 Implementation.....	43
5.4 Methods of the Profile	46
5.5 Use Cases.....	46
5.6 CIM Elements.....	46
6 FC Target Ports Profile	49
6.1 Synopsis.....	49
6.2 Description	49
6.3 Implementation.....	49
6.4 Durable Names and Correlatable IDs of the Profile	50
6.5 Health and Fault Management.....	50
6.6 Supported Profiles and Packages.....	50
6.7 Extrinsic Methods of this Profile	50
6.8 Client Considerations and Recipes	50
6.9 CIM Elements.....	50
7 FCoE Target Ports Profile	55
7.1 Synopsis.....	55
7.2 Description	55
7.3 Implementation.....	56
7.4 Durable Names and Correlatable IDs of the Profile	56
7.5 Methods	57
7.6 Use Cases.....	57
7.7 CIM Elements.....	57
8 iSCSI Target Ports Profile	63
8.1 Synopsis.....	63
8.2 Description	63
8.3 Implementation.....	63
8.4 Health and Fault Management.....	67
8.5 Methods of this Profile.....	67
8.6 Client Considerations and Recipes	71
8.7 CIM Elements.....	72

9	Serial Attached SCSI (SAS) Target Port Profile	93
9.1	Synopsis.....	93
9.2	Description	93
9.3	Methods	94
9.4	Client Considerations and Recipes	94
9.5	CIM Elements.....	95
10	Serial ATA (SATA) Target Ports Profile	99
11	SB Target Ports Profile.....	101
11.1	Synopsis.....	101
11.2	Description	101
11.3	Implementation.....	101
11.4	Health and Fault Management Consideration.....	102
11.5	Cascading Considerations	103
11.6	Methods of the Profile	103
11.7	Client Considerations and Recipes	103
11.8	CIM Elements.....	103
12	Direct Attach (DA) Ports Profile	107
12.1	Synopsis.....	107
12.2	Description	107
12.3	Health and Fault Management	108
12.4	Extrinsic Methods.....	108
12.5	Use Cases.....	108
12.6	CIM Elements.....	108
13	Generic Initiator Ports Profile.....	113
13.1	Synopsis.....	113
13.2	Description	113
13.3	Implementation.....	113
13.4	Methods	118
13.5	Use Cases.....	119
13.6	CIM Elements.....	119
14	Parallel SCSI (SPI) Initiator Ports Profile	125
14.1	Synopsis	125
14.2	Description.....	125
14.3	Implementation	125
14.4	Methods	126
14.5	Use Cases and Recipes	126
14.6	CIM Elements	126
15	iSCSI Initiator Port Profile.....	135
15.1	Synopsis.....	135
15.2	Description	135
15.3	Implementation.....	135
15.4	Methods	137
15.5	Use Cases and Recipes.....	137
15.6	CIM Elements.....	137
16	FC Initiator Ports Profile	143
16.1	Synopsis.....	143
16.2	Description	143
16.3	Implementation.....	143
16.4	Methods	145

16.5	Use Cases and Recipes.....	145
16.6	CIM Elements.....	145
17	SAS Initiator Ports Profile	155
17.1	Synopsis.....	155
17.2	Description	155
17.3	Methods of the profile.....	156
17.4	Client Considerations and Recipes	156
17.5	CIM Elements.....	156
18	ATA Initiator Ports Profile.....	167
19	FC-SB-x Initiator Ports Profile	169
19.1	Synopsis.....	169
19.2	Description	169
19.3	Implementation.....	169
19.4	Methods	170
19.5	Client Considerations and Recipes	170
19.6	CIM Elements.....	171
20	FCoE Initiator Ports Profile	179
20.1	Synopsis.....	179
20.2	Description	179
20.3	Implementation.....	179
20.4	Methods	181
20.5	Use Cases and Recipes.....	182
20.6	CIM Elements.....	182
21	Access Points Profile.....	193
21.1	Synopsis.....	193
21.2	Description	193
21.3	Health and Fault Management Considerations	194
21.4	Cascading Considerations	194
21.5	Methods of this Profile.....	194
21.6	Client Considerations and Recipes	195
21.7	CIM Elements.....	195
22	Health Package	197
22.1	Synopsis.....	197
22.2	Description	197
22.3	Health and Fault Management Considerations	201
22.4	Cascading Considerations	201
22.5	Use Cases.....	201
22.6	CIM Elements.....	202
23	Job Control Profile	205
23.1	Synopsis.....	205
23.2	Description	205
23.3	Health and Fault Management	208
23.4	Cascading Considerations	209
23.5	Methods of the Profile	209
23.6	Client Considerations and Recipes	210
23.7	Registered Name and Version	211
23.8	CIM Elements.....	211
24	Location Profile.....	217
24.1	Synopsis.....	217

24.2	Description	217
24.3	Health and Fault Management Considerations	217
24.4	Cascading Considerations	217
24.5	Methods of the Profile	217
24.6	Use Cases.....	218
24.7	CIM Elements.....	218
25	Multiple Computer System Profile	221
25.1	Synopsis.....	221
25.2	Description	221
25.3	Health and Fault Management Considerations	225
25.4	Cascading Considerations	225
25.5	Methods of the Profile	225
25.6	Use Cases.....	226
25.7	CIM Elements.....	226
26	Physical Package Package	231
26.1	Synopsis.....	231
26.2	Description	231
26.3	Health and Fault Management Considerations	233
26.4	Cascading Considerations	233
26.5	Methods of this Profile.....	233
26.6	Use Cases.....	233
26.7	CIM Elements.....	234
27	Power Supply Profile	241
27.1	Synopsis.....	241
27.2	Description	241
27.3	Implementation.....	241
27.4	Methods	241
27.5	Use Cases.....	241
27.6	CIM Elements.....	242
28	Fan Profile	247
28.1	Synopsis.....	247
28.2	Description	247
28.3	Implementation.....	247
28.4	Methods	247
28.5	Use Cases.....	247
28.6	CIM Elements.....	248
29	Sensors Profile	255
29.1	Synopsis.....	255
29.2	Description	255
29.3	Implementation.....	255
29.4	Methods	255
29.5	Use Cases.....	255
29.6	CIM Elements.....	256
30	Base Server Profile.....	263
30.1	Synopsis.....	263
30.2	Description	264
30.3	Implementation.....	264
30.4	Methods	264
30.5	Use Cases.....	264
30.6	CIM Elements.....	264

31	Media Access Device Profile	269
31.1	Synopsis.....	269
31.2	Description	269
31.3	Implementation.....	270
31.4	Methods	271
31.5	Use Cases.....	271
31.6	CIM Elements.....	271
32	Storage Enclosure Profile.....	277
32.1	Synopsis.....	277
32.2	Description	277
32.3	Implementation.....	280
32.4	Methods	282
32.5	Use Cases.....	283
32.6	CIM Elements.....	283
33	Software Profile	285
33.1	Synopsis.....	285
33.2	Description	285
33.3	Health and Fault Management Considerations	286
33.4	Cascading Considerations	286
33.5	Methods of the Profile	286
33.6	Use Cases.....	286
33.7	CIM Elements.....	286
34	Software Inventory Profile	289
34.1	Synopsis.....	289
34.2	Description	289
34.3	Implementation.....	290
34.4	Methods	290
34.5	Use Cases.....	290
34.6	CIM Elements.....	290
35	Server Profile	297
36	Profile Registration Profile	299
36.1	Synopsis.....	299
36.2	Description	299
36.3	Implementation.....	299
36.4	Methods	302
36.5	Use Cases.....	302
36.6	CIM Elements.....	303
37	Indication Profile	309
38	Object Manager Adapter Profile	311
39	Proxy Server System Management Profile.....	313
40	Device Credentials Profile	315
40.1	Synopsis.....	315
40.2	Description	315
40.3	Health and Fault Management Considerations	315
40.4	Cascading Considerations	316
40.5	Extrinsic Methods of this Profile	316
40.6	Use Cases.....	316
40.7	CIM Elements.....	316
41	Operational Power Profile.....	319

41.1	Synopsis.....	319
41.2	Description	319
41.3	Implementation.....	320
41.4	Methods of the Profile	326
41.5	Use Cases.....	331
41.6	Client Considerations and Recipes	331
41.7	CIM Elements.....	331
42	Indications Profile	347
43	WBEM Server Profile.....	349
	Annex A (informative) SMI-S Information Model.....	351
	Annex B (informative) Cross Profile Considerations	353
B.1	Overview	353
B.2	HBA model	353
B.3	Switch Model	354
B.4	Array Model	354
B.5	Storage Virtualization Model	356
B.6	Fabric Topology (HBA, Switch, Array)	357

LIST OF FIGURES

Figure 1 - Experimental Maturity Level Tag	10
Figure 2 - Implemented Maturity Level Tag	10
Figure 3 - Stable Maturity Level Tag	11
Figure 4 - Deprecated Tag	11
Figure 5 - Generic Target Port Classes	43
Figure 6 - LogicalPort Class Hierarchy	44
Figure 7 - Generic Target with LUN Masking	45
Figure 8 - FC Target Port Instance Diagram	49
Figure 9 - FCoE Topology	55
Figure 10 - EthernetPort used for FCoE	56
Figure 11 - iSCSI Target Ports Profile Instance Diagram	65
Figure 12 - Serial Attached SCSI (SAS) Target Port Instance Diagram	93
Figure 13 - SB Target Port Instance Diagram	102
Figure 14 - DA Port Instance Diagram	108
Figure 15 - Generic Initiator Port Model	114
Figure 16 - Optional Connectivity Collection Model	114
Figure 17 - Optional Full-Path Model	115
Figure 18 - HBA and Disk Model	116
Figure 19 - HBA and Tape or Optical Devices	117
Figure 20 - Port Statistics	117
Figure 21 - Port Statistics Hierarchy	118
Figure 22 - SPI Initiator Port Instance Diagram	125
Figure 23 - iSCSI Initiator Port Instance Diagram	136
Figure 24 - Fibre Channel Initiator Instance Diagram	144
Figure 25 - FC Node Model	144
Figure 26 - SAS Initiator Port Model	155
Figure 27 - Fibre Channel Initiator Instance Diagram	169
Figure 28 - FCoE Initiator Instance Diagram	179
Figure 29 - Optional Target Element Model	180
Figure 30 - Logical Port Group Model	181
Figure 31 - System-wide Remote Access Point	193
Figure 32 - Access Point Instance Diagram	194
Figure 33 - Job Control Profile Model	205
Figure 34 - Storage Configuration	211
Figure 35 - Location Instance	217
Figure 36 - Two Redundant Systems Instance Diagram	221
Figure 37 - Multiple Redundancy Tier Instance Diagram	223
Figure 38 - System Level Numbers	224
Figure 39 - Physical Package Package Mandatory Classes	231
Figure 40 - Modeling for well defined subcomponents	232
Figure 41 - Physical Package Package with Optional Classes	233
Figure 42 - Media Access Device Class Diagram	270
Figure 43 - Enclosure with Two Arrays	279
Figure 44 - Model for Disk in Enclosure	282

Figure 45 - Software Instance Diagram	285
Figure 46 - Associations between RegisteredProfile instances	300
Figure 47 - Model for SMI-S Registered Profile	301
Figure 48 - Model for Provider Versions	301
Figure 49 - DeviceCredentials Profile Model.....	315
Figure 50 - Operational Power Profile Summary	320
Figure 51 - Model for Element Types.....	321
Figure 52 - Classes related to Top-level System Power Statistics.....	322

LIST OF TABLES

Table 1 - Profile Components	41
Table 2 - Modeling of Common Storage Devices in CIM.....	45
Table 3 - CIM Elements for Generic Target Ports	46
Table 4 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	47
Table 5 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint	47
Table 6 - SMI Referenced Properties/Methods for CIM_LogicalPort.....	47
Table 7 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint	48
Table 8 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port).....	48
Table 9 - Related Profiles for FC Target Ports	49
Table 10 - FCPort OperationalStatus	50
Table 11 - CIM Elements for FC Target Ports	50
Table 12 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	51
Table 13 - SMI Referenced Properties/Methods for CIM_FCPort	51
Table 14 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint	52
Table 15 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint.....	53
Table 16 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port).....	53
Table 17 - Related Profiles for FCoE Target Ports	55
Table 18 - FCPort OperationalStatus	56
Table 19 - CIM Elements for FCoE Target Ports.....	57
Table 20 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	58
Table 21 - SMI Referenced Properties/Methods for CIM_EthernetPort	58
Table 22 - SMI Referenced Properties/Methods for CIM_FCPort (For FCoE)	58
Table 23 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint	59
Table 24 - SMI Referenced Properties/Methods for CIM_HostedDependency (NetworkPort to FCPort)	60
Table 25 - SMI Referenced Properties/Methods for CIM_LogicalPort.....	60
Table 26 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint.....	61
Table 27 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port).....	61
Table 28 - iSCSI Terminology and SMI-S Class Names	63
Table 29 - EthernetPort OperationalStatus.....	67
Table 30 - CIM Elements for iSCSI Target Ports.....	72
Table 31 - SMI Referenced Properties/Methods for CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint).....	74
Table 32 - SMI Referenced Properties/Methods for CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)	75
Table 33 - SMI Referenced Properties/Methods for CIM_ConcreteDependency.....	75
Table 34 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (EthernetPort to IPProtocolEndpoint).....	75
Table 35 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (EthernetPort to iSCSIProtocolEndpoint).....	76
Table 36 - SMI Referenced Properties/Methods for CIM_ElementCapabilities (CIM_iSCSICapabilities to ComputerSystem).....	76
Table 37 - SMI Referenced Properties/Methods for CIM_ElementCapabilities (iSCSIConfigurationCapabilities to iSCSIConfigurationService)	76
Table 38 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSIConnectionSettings to TCPProtocolEndpoint).....	77
Table 39 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint).....	77
Table 40 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolController)	77

Table 41 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSI Session Settings to ComputerSystem).....	78
Table 42 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSI Session Settings to iSCSI Protocol Endpoint).....	78
Table 43 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (iSCSI Login Statistics to SCSI Protocol Controller)	78
Table 44 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (iSCSI Session Failures to SCSI Protocol Controller)	79
Table 45 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (iSCSI Session Statistics to iSCSI Session)	79
Table 46 - SMI Referenced Properties/Methods for CIM_EndpointOfNetworkPipe (iSCSI Connection to TCP Protocol Endpoint).....	79
Table 47 - SMI Referenced Properties/Methods for CIM_EndpointOfNetworkPipe (iSCSI Session to iSCSI Protocol Endpoint).....	80
Table 48 - SMI Referenced Properties/Methods for CIM_EthernetPort	80
Table 49 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to IP Protocol Endpoint).....	81
Table 50 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to TCP Protocol Endpoint).....	81
Table 51 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to iSCSI Protocol Endpoint).....	81
Table 52 - SMI Referenced Properties/Methods for CIM_HostedCollection	82
Table 53 - SMI Referenced Properties/Methods for CIM_HostedService	82
Table 54 - SMI Referenced Properties/Methods for CIM_IPProtocolEndpoint.....	82
Table 55 - SMI Referenced Properties/Methods for CIM_iSCSICapabilities.....	83
Table 56 - SMI Referenced Properties/Methods for CIM_iSCSIConfigurationCapabilities	83
Table 57 - SMI Referenced Properties/Methods for CIM_iSCSIConfigurationService	84
Table 58 - SMI Referenced Properties/Methods for CIM_iSCSIConnection	84
Table 59 - SMI Referenced Properties/Methods for CIM_iSCSIConnectionSettings	85
Table 60 - SMI Referenced Properties/Methods for CIM_iSCSILoginStatistics	85
Table 61 - SMI Referenced Properties/Methods for CIM_iSCSIProtocolEndpoint	86
Table 62 - SMI Referenced Properties/Methods for CIM_iSCSI Session.....	86
Table 63 - SMI Referenced Properties/Methods for CIM_iSCSI Session Failures.....	87
Table 64 - SMI Referenced Properties/Methods for CIM_iSCSI Session Settings.....	88
Table 65 - SMI Referenced Properties/Methods for CIM_iSCSI Session Statistics	88
Table 66 - SMI Referenced Properties/Methods for CIM_MemberOfCollection	89
Table 67 - SMI Referenced Properties/Methods for CIM_NetworkPipeComposition	89
Table 68 - SMI Referenced Properties/Methods for CIM_SAP Available For Element	89
Table 69 - SMI Referenced Properties/Methods for CIM_SCSI Protocol Controller	90
Table 70 - SMI Referenced Properties/Methods for CIM_System Device (ComputerSystem to Ethernet Port)	90
Table 71 - SMI Referenced Properties/Methods for CIM_System Device (ComputerSystem to SCSI Protocol Controller).....	90
Table 72 - SMI Referenced Properties/Methods for CIM_System Specific Collection	91
Table 73 - SMI Referenced Properties/Methods for CIM_TCP Protocol Endpoint	91
Table 74 - Related Profiles for SAS Target Ports	93
Table 75 - SAS Port Operational Status	94
Table 76 - CIM Elements for SAS Target Ports.....	95
Table 77 - SMI Referenced Properties/Methods for CIM_Concrete Component	95
Table 78 - SMI Referenced Properties/Methods for CIM_Device SAP Implementation	96
Table 79 - SMI Referenced Properties/Methods for CIM_Hosted Access Point	96
Table 80 - SMI Referenced Properties/Methods for CIM_SAS Port.....	96

Table 81 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint.....	97
Table 82 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port).....	97
Table 83 - SMI Referenced Properties/Methods for CIM_SystemDevice (SAS PHY).....	98
Table 84 - SMI Referenced Properties/Methods for CIM_SASPHY	98
Table 85 - Related Profiles for SB Target Ports	101
Table 86 - FCPort OperationalStatus	102
Table 87 - CIM Elements for SB Target Ports	103
Table 88 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	104
Table 89 - SMI Referenced Properties/Methods for CIM_FCPort	104
Table 90 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint	105
Table 91 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port).....	105
Table 92 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint	106
Table 93 - CIM Elements for DA Target Ports	108
Table 94 - SMI Referenced Properties/Methods for CIM_DAPort	109
Table 95 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	109
Table 96 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint	110
Table 97 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint.....	110
Table 98 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port).....	111
Table 99 - Related Profiles for Generic Initiator Ports	113
Table 100 - CIM Elements for Generic Initiator Ports	119
Table 101 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection	119
Table 102 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	120
Table 103 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)	120
Table 104 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)	120
Table 105 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target).....	121
Table 106 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection).....	121
Table 107 - SMI Referenced Properties/Methods for CIM_LogicalPort.....	122
Table 108 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)	122
Table 109 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Initiator)	122
Table 110 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Target).....	123
Table 111 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)	124
Table 112 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics	124
Table 113 - SPIPort OperationalStatus.....	125
Table 114 - CIM Elements for SPI Initiator Ports	126
Table 115 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection	127
Table 116 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	128
Table 117 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)	128
Table 118 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)	128
Table 119 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target).....	129
Table 120 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection).....	129
Table 121 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)	129
Table 122 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath	130
Table 123 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator).....	130
Table 124 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)	131
Table 125 - SMI Referenced Properties/Methods for CIM_SPIPort	131
Table 126 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)	132
Table 127 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics	132
Table 128 - Related Profiles for iSCSI Initiator Ports.....	135
Table 129 - EthernetPort OperationalStatus.....	136

Table 130 - CIM Elements for iSCSI Initiator Ports.....	137
Table 131 - SMI Referenced Properties/Methods for CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)	138
Table 132 - SMI Referenced Properties/Methods for CIM_BindsTo(TCPProtocolEndpoint to IPProtocolEndpoint)	138
Table 133 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (IPProtocolEndpoint to EthernetPort)	139
Table 134 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (iSCSIProtocolEndpoint to EthernetPort)	139
Table 135 - SMI Referenced Properties/Methods for CIM_EthernetPort	139
Table 136 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint).....	140
Table 137 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint).....	140
Table 138 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint).....	140
Table 139 - SMI Referenced Properties/Methods for CIM_IPProtocolEndpoint	141
Table 140 - SMI Referenced Properties/Methods for CIM_iSCSIProtocolEndpoint	141
Table 141 - SMI Referenced Properties/Methods for CIM_SystemDevice (ComputerSystem to EthernetPort) .	142
Table 142 - SMI Referenced Properties/Methods for CIM_TCPProtocolEndpoint	142
Table 143 - Related Profiles for FC Initiator Ports	143
Table 144 - FCPort OperationalStatus.....	145
Table 145 - CIM Elements for FC Initiator Ports	145
Table 146 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection	146
Table 147 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation.....	147
Table 148 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)	147
Table 149 - SMI Referenced Properties/Methods for CIM_FCPort	148
Table 150 - SMI Referenced Properties/Methods for CIM_FCPortStatistics	149
Table 151 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)	149
Table 152 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target).....	150
Table 153 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection).....	150
Table 154 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)	151
Table 155 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath	151
Table 156 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator).....	151
Table 157 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)	152
Table 158 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)	153
Table 159 - SASPort OperationalStatus	156
Table 160 - CIM Elements for SAS Initiator Ports.....	156
Table 161 - SMI Referenced Properties/Methods for CIM_ATAProtocolEndpoint (Initiator)	157
Table 162 - SMI Referenced Properties/Methods for CIM_BindsTo.....	158
Table 163 - SMI Referenced Properties/Methods for CIM_ConcreteComponent.....	158
Table 164 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection	158
Table 165 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	159
Table 166 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (PHY Statistics)	159
Table 167 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)	159
Table 168 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)	160
Table 169 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target).....	160
Table 170 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection).....	160
Table 171 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)	161
Table 172 - SMI Referenced Properties/Methods for CIM_SASPort.....	161

Table 173 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath	162
Table 174 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator)	162
Table 175 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)	163
Table 176 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator PHY)	163
Table 177 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)	164
Table 178 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics	164
Table 179 - SMI Referenced Properties/Methods for CIM_SASPHY	164
Table 180 - SMI Referenced Properties/Methods for CIM_SASPhyStatistics	165
Table 181 - Related Profiles for SB Initiator Ports	169
Table 182 - FCPort OperationalStatus.....	170
Table 183 - CIM Elements for SB Initiator Ports	171
Table 184 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection	172
Table 185 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	172
Table 186 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)	172
Table 187 - SMI Referenced Properties/Methods for CIM_FCPort	173
Table 188 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)	174
Table 189 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target).....	174
Table 190 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection).....	174
Table 191 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)	175
Table 192 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)	175
Table 193 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics	175
Table 194 - SMI Referenced Properties/Methods for CIM_SBInitiatorTargetLogicalUnitPath.....	176
Table 195 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint (Initiator)	176
Table 196 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint (Target).....	177
Table 197 - FCPort OperationalStatus.....	181
Table 198 - CIM Elements for FCoE Initiator Ports.....	182
Table 199 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection	183
Table 200 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation	184
Table 201 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)	184
Table 202 - SMI Referenced Properties/Methods for CIM_EthernetPort.....	184
Table 203 - SMI Referenced Properties/Methods for CIM_FCPort	185
Table 204 - SMI Referenced Properties/Methods for CIM_FCPortStatistics	186
Table 205 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)	186
Table 206 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target).....	187
Table 207 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection).....	187
Table 208 - SMI Referenced Properties/Methods for CIM_HostedCollection (FC Node).....	187
Table 209 - SMI Referenced Properties/Methods for CIM_HostedDependency (NetworkPort to FCPort).....	188
Table 210 - SMI Referenced Properties/Methods for CIM_LogicalPortGroup.....	188
Table 211 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)	189
Table 212 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (FC Node)	189
Table 213 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Initiator)	189
Table 214 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Target).....	190
Table 215 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath	190
Table 216 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator).....	191
Table 217 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)	191
Table 218 - SMI Referenced Properties/Methods for CIM_SystemDevice (Ethernet Port)	192
Table 219 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)	192
Table 220 - RemoteAccessPoint InfoFormat and AccessInfo Properties	194
Table 221 - CIM Elements for Access Points	195

Table 222 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint.....	195
Table 223 - SMI Referenced Properties/Methods for CIM_RemoteServiceAccessPoint	195
Table 224 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement	196
Table 225 - OperationalStatus Details	199
Table 226 - CIM Elements for Health.....	202
Table 227 - SMI Referenced Properties/Methods for CIM_ComputerSystem	202
Table 228 - SMI Referenced Properties/Methods for CIM_LogicalDevice	203
Table 229 - SMI Referenced Properties/Methods for CIM_RelatedElementCausingError	203
Table 230 - OperationalStatus to Job State Mapping.....	208
Table 231 - Standard Message for Job Control Profile.....	209
Table 232 - CIM Elements for Job Control.....	211
Table 233 - SMI Referenced Properties/Methods for CIM_AffectedJobElement.....	212
Table 234 - SMI Referenced Properties/Methods for CIM_AssociatedJobMethodResult	213
Table 235 - SMI Referenced Properties/Methods for CIM_ConcreteJob	213
Table 236 - SMI Referenced Properties/Methods for CIM_MethodResult.....	214
Table 237 - SMI Referenced Properties/Methods for CIM_OwningJobElement.....	215
Table 238 - CIM Elements for Location	218
Table 239 - SMI Referenced Properties/Methods for CIM_Location	218
Table 240 - SMI Referenced Properties/Methods for CIM_PhysicalElementLocation.....	218
Table 241 - Related Profiles for Multiple Computer System	221
Table 242 - Redundancy Type.....	222
Table 243 - CIM Elements for Multiple Computer System.....	226
Table 244 - SMI Referenced Properties/Methods for CIM_ComponentCS	227
Table 245 - SMI Referenced Properties/Methods for CIM_ComputerSystem (Non-Top-Level System)	227
Table 246 - SMI Referenced Properties/Methods for CIM_ConcreteIdentity.....	227
Table 247 - SMI Referenced Properties/Methods for CIM_IsSpare	228
Table 248 - SMI Referenced Properties/Methods for CIM_MemberOfCollection	228
Table 249 - SMI Referenced Properties/Methods for CIM_RedundancySet	229
Table 250 - CIM Elements for Physical Package.....	234
Table 251 - SMI Referenced Properties/Methods for CIM_Container	235
Table 252 - SMI Referenced Properties/Methods for CIM_LogicalIdentity.....	235
Table 253 - SMI Referenced Properties/Methods for CIM_PhysicalElementLocation.....	235
Table 254 - SMI Referenced Properties/Methods for CIM_PhysicalPackage (Component)	236
Table 255 - SMI Referenced Properties/Methods for CIM_PhysicalPackage (System).....	236
Table 256 - SMI Referenced Properties/Methods for CIM_Product (Component)	237
Table 257 - SMI Referenced Properties/Methods for CIM_Product (System)	237
Table 258 - SMI Referenced Properties/Methods for CIM_ProductParentChild.....	238
Table 259 - SMI Referenced Properties/Methods for CIM_ProductPhysicalComponent (Component)	238
Table 260 - SMI Referenced Properties/Methods for CIM_ProductPhysicalComponent (System)	239
Table 261 - SMI Referenced Properties/Methods for CIM_SystemPackaging (Component)	239
Table 262 - SMI Referenced Properties/Methods for CIM_SystemPackaging (System).....	239
Table 263 - Related Profiles for Power Supply	241
Table 264 - CIM Elements for Power Supply.....	242
Table 265 - SMI Referenced Properties/Methods for CIM_ElementCapabilities	243
Table 266 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities	243
Table 267 - SMI Referenced Properties/Methods for CIM_IsSpare	243
Table 268 - SMI Referenced Properties/Methods for CIM_MemberOfCollection	244
Table 269 - SMI Referenced Properties/Methods for CIM_OwningCollectionElement.....	244
Table 270 - SMI Referenced Properties/Methods for CIM_PowerSupply.....	245

Table 271 - SMI Referenced Properties/Methods for CIM_RedundancySet	245
Table 272 - SMI Referenced Properties/Methods for CIM_SuppliesPower	246
Table 273 - SMI Referenced Properties/Methods for CIM_SystemDevice	246
Table 274 - Related Profiles for Fan	247
Table 275 - CIM Elements for Fan	248
Table 276 - SMI Referenced Properties/Methods for CIM_AssociatedCooling	249
Table 277 - SMI Referenced Properties/Methods for CIM_AssociatedSensor	249
Table 278 - SMI Referenced Properties/Methods for CIM_ElementCapabilities	249
Table 279 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities	250
Table 280 - SMI Referenced Properties/Methods for CIM_Fan	250
Table 281 - SMI Referenced Properties/Methods for CIM_HostedCollection	251
Table 282 - SMI Referenced Properties/Methods for CIM_IsSpare	251
Table 283 - SMI Referenced Properties/Methods for CIM_MemberOfCollection	252
Table 284 - SMI Referenced Properties/Methods for CIM_NumericSensor	252
Table 285 - SMI Referenced Properties/Methods for CIM_OwningCollectionElement	253
Table 286 - SMI Referenced Properties/Methods for CIM_RedundancySet (Fan Redundancy)	253
Table 287 - SMI Referenced Properties/Methods for CIM_Sensor	253
Table 288 - SMI Referenced Properties/Methods for CIM_SystemDevice	254
Table 289 - Related Profiles for Fan	255
Table 290 - CIM Elements for Sensors	256
Table 291 - SMI Referenced Properties/Methods for CIM_AssociatedSensor	257
Table 292 - SMI Referenced Properties/Methods for CIM_ElementCapabilities	257
Table 293 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities	257
Table 294 - SMI Referenced Properties/Methods for CIM_NumericSensor	258
Table 295 - SMI Referenced Properties/Methods for CIM_Sensor	260
Table 296 - SMI Referenced Properties/Methods for CIM_SystemDevice	261
Table 297 - Related Profiles for Base Server	263
Table 298 - CIM Elements for Base Server	264
Table 299 - SMI Referenced Properties/Methods for CIM_ComputerSystem	265
Table 300 - SMI Referenced Properties/Methods for CIM_ComputerSystemPackage	266
Table 301 - SMI Referenced Properties/Methods for CIM_ElementCapabilities	266
Table 302 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities	266
Table 303 - SMI Referenced Properties/Methods for CIM_HostedService	267
Table 304 - SMI Referenced Properties/Methods for CIM_PhysicalPackage	267
Table 305 - SMI Referenced Properties/Methods for CIM_ServiceAffectsElement	268
Table 306 - SMI Referenced Properties/Methods for CIM_TimeService	268
Table 307 - Related Profiles for Media Access Device	269
Table 308 - OperationalStatus For MediaAccessDevice	270
Table 309 - CIM Elements for Media Access Device	271
Table 310 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities	272
Table 311 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint	272
Table 312 - SMI Referenced Properties/Methods for CIM_MediaAccessDevice	272
Table 313 - SMI Referenced Properties/Methods for CIM_PhysicalPackage	273
Table 314 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint	274
Table 315 - SMI Referenced Properties/Methods for CIM_Realizes	274
Table 316 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement	274
Table 317 - SMI Referenced Properties/Methods for CIM_SystemDevice	275
Table 318 - Related Profiles for Storage Enclosure	277
Table 319 - CIM Elements for Storage Enclosure	283

Table 320 - SMI Referenced Properties/Methods for CIM_ConfigurationReportingService.....	284
Table 321 - SMI Referenced Properties/Methods for CIM_HostedService	284
Table 322 - CIM Elements for Software.....	286
Table 323 - SMI Referenced Properties/Methods for CIM_InstalledSoftwareIdentity.....	286
Table 324 - SMI Referenced Properties/Methods for CIM_SoftwareIdentity	287
Table 325 - Related Profiles for Software Inventory	289
Table 326 - CIM Elements for Software Inventory	290
Table 327 - SMI Referenced Properties/Methods for CIM_ElementSoftwareIdentity.....	291
Table 328 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint.....	291
Table 329 - SMI Referenced Properties/Methods for CIM_HostedCollection.....	292
Table 330 - SMI Referenced Properties/Methods for CIM_InstalledSoftwareIdentity.....	292
Table 331 - SMI Referenced Properties/Methods for CIM_MemberOfCollection	292
Table 332 - SMI Referenced Properties/Methods for CIM_OrderedComponent.....	293
Table 333 - SMI Referenced Properties/Methods for CIM_OrderedDependency	293
Table 334 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement	293
Table 335 - SMI Referenced Properties/Methods for CIM_SoftwareIdentity.....	294
Table 336 - SMI Referenced Properties/Methods for CIM_SoftwareIdentityResource.....	294
Table 337 - SMI Referenced Properties/Methods for CIM_SystemSpecificCollection	295
Table 338 - CIM Elements for Profile Registration.....	303
Table 339 - SMI Referenced Properties/Methods for CIM_ElementConformsToProfile (Associates Domain object (e.g. System) to RegisteredProfile).....	304
Table 340 - SMI Referenced Properties/Methods for CIM_ElementConformsToProfile (Associates Regis- teredProfiles for SMI-S and domain profiles).....	304
Table 341 - SMI Referenced Properties/Methods for CIM_ElementSoftwareIdentity (Profile and SW identity) ..	305
Table 342 - SMI Referenced Properties/Methods for CIM_ElementSoftwareIdentity (Profile and SW identity) ..	305
Table 343 - SMI Referenced Properties/Methods for CIM_Product	305
Table 344 - SMI Referenced Properties/Methods for CIM_ProductSoftwareComponent.....	306
Table 345 - SMI Referenced Properties/Methods for CIM_ReferencedProfile.....	306
Table 346 - SMI Referenced Properties/Methods for CIM_RegisteredProfile (Domain Registered Profile).....	306
Table 347 - SMI Referenced Properties/Methods for CIM_RegisteredProfile (The SMI-S Registered Profile) ..	307
Table 348 - SMI Referenced Properties/Methods for CIM_SoftwareIdentity.....	308
Table 349 - CIM Elements for Device Credentials.....	316
Table 350 - SMI Referenced Properties/Methods for CIM_HostedService	316
Table 351 - SMI Referenced Properties/Methods for CIM_SharedSecret.....	316
Table 352 - SMI Referenced Properties/Methods for CIM_SharedSecretIsShared	317
Table 353 - SMI Referenced Properties/Methods for CIM_SharedSecretService.....	317
Table 354 - Related Profiles for Operational Power	319
Table 355 - Creation, Deletion and Modification Methods	326
Table 356 - CIM Elements for Operational Power	331
Table 357 - SMI Referenced Properties/Methods for CIM_ElementCapabilities	333
Table 358 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Component System Stats) ..	334
Table 359 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Top Level System Stats) ..	334
Table 360 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Volume Stats).....	335
Table 361 - SMI Referenced Properties/Methods for CIM_HostedCollection (Client Defined).....	335
Table 362 - SMI Referenced Properties/Methods for CIM_HostedCollection (Default).....	336
Table 363 - SMI Referenced Properties/Methods for CIM_HostedCollection (Systemto StatisticsCollection) ..	336
Table 364 - SMI Referenced Properties/Methods for CIM_HostedService	336
Table 365 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (DeviceSet)	337
Table 366 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Member of client defined collection)	337

Table 367 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Member of pre-defined collection)	337
Table 368 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Member of statistics collection).....	338
Table 369 - SMI Referenced Properties/Methods for CIM_StatisticsCollection.....	338
Table 370 - SMI Referenced Properties/Methods for CIM_DeviceSet (Provider Defined)	339
Table 371 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifest (Client Defined)	339
Table 372 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifest (Provider Support)	340
Table 373 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifestCollection (Client Defined).....	341
Table 374 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifestCollection (Provider Defined).....	342
Table 375 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticalData	342
Table 376 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticsCapabilities	343
Table 377 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticsService.....	345

FOREWORD

The Storage Management Technical Specification is published in several parts. *Storage Management Technical Specification, Part 3 Common Profiles, 1.7.0 Rev 5* defines profiles that are used by profiles in other parts of this standard. In general, the common profiles do not fully define storage elements, but define non-storage management aspects that are common to storage domains. For example, the Access Points Profile defines a technique that the arrays, switches, or libraries may use to inform clients of non-CIM network interfaces that are available.

Some of the common profiles are based on DMTF profiles. For these profiles, the DMTF profile may be “specialized” to assure SNIA requirements are met.

Parts of this Standard

This standard is subdivided in the following parts:

- *Storage Management Technical Specification, Part 1 Overview, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 3 Common Profiles, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 4 Block Devices, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 5 Filesystems, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 6 Fabric, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 7 Host Elements, 1.7.0 Rev 5*
- *Storage Management Technical Specification, Part 8 Media Libraries, 1.7.0 Rev 5*

SNIA Web Site

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

SNIA Address

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent via the SNIA Feedback Portal at <http://www.snia.org/feedback/> or by mail to the Storage Networking Industry Association, 4360 ArrowsWest Drive, Colorado Springs, Colorado 80907, U.S.A.

1 Scope

Storage Management Technical Specification, Part 3 Common Profiles, 1.7.0 Rev 5 defines profiles that are supported by profiles defined in the other parts of this standard. The first few clauses provide background material that helps explain the purpose and profiles. Common port profiles are grouped together since they serve as transport-specific variations of a common model. The port profiles are followed by other common profiles.

2 Normative References

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

2.1 Approved References

ISO/IEC 14776-413, SCSI Architecture Model - 3 (SAM-3) [ANSI INCITS 402-200x]

ISO/IEC 14776-452, SCSI Primary Commands - 3 (SPC-3) [ANSI INCITS.351-2005]

ANSI/INCITS 374:2003, Information technology - Fibre Channel Single - Byte Command Set-3 (FC-SB-3)

2.2 DMTF References (Final)

DMTF Final documents are accepted as standards.

DMTF DSP1001 Management Profile Specification Usage Guide 1.2.0

http://www.dmtf.org/sites/default/files/standards/documents/DSP1001_1.2.0.pdf

DMTF DSP1004 Base Server Profile 1.0.1

http://www.dmtf.org/sites/default/files/standards/documents/DSP1004_1.0.1.pdf

DMTF DSP1009 Sensors Profile, 1.1.1

http://www.dmtf.org/sites/default/files/standards/documents/DSP1009_1.1.1.pdf

DMTF DSP1011 Physical Asset Profile 1.0.2

http://www.dmtf.org/standards/published_documents/DSP1011_1.0.2.pdf

DMTF DSP1013:2006, Fan Profile 1.0.1

http://www.dmtf.org/standards/published_documents/DSP1013_1.0.1.pdf

DMTF DSP1015 Power Supply Profile 1.1.0

http://www.dmtf.org/standards/published_documents/DSP1015_1.1.0.pdf

DMTF DSP1025 Software Update Profile 1.0.0

http://www.dmtf.org/standards/published_documents/DSP1025_1.0.0.pdf

DMTF DSP1033 Profile Registration Profile 1.1.0

http://www.dmtf.org/sites/default/files/standards/documents/DSP1033_1.1.0.pdf

DMTF DPS1052 Computer System Profile 1.0.3

http://www.dmtf.org/sites/default/files/standards/documents/DSP1052_1.0.3.pdf

DMTF DSP1054 Indications Profile 1.2.2

http://www.dmtf.org/sites/default/files/standards/documents/DSP1054_1.2.2.pdf

DMTF DSP1092:2013, WBEM Server Profile 1.0

http://www.dmtf.org/standards/published_documents/DSP1092_1.0.pdf

DMTF DSP1102 Launch in Context Profile 1.0.0

www.dmtf.org/sites/default/files/standards/documents/DSP1102_1.0.0_0.pdf

DMTF DSP1103 Job Control Profile v1.0.0

http://www.dmtf.org/sites/default/files/standards/documents/DSP1103_1.0.0_0.pdf

2.3 References under development

DMTF DSP1119, Diagnostics Job Control Profile v1.0.0b

http://www.dmtf.org/sites/default/files/standards/documents/DSP1119_1.0.0b.pdf

Normative References

Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5

3 Terms and Definitions

3.1 General

For the purposes of this document, the definitions, symbols, abbreviations, and conventions given in *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5* and the following apply.

3.2 Terms

3.2.1

FC-SB-X

Fibre Channel Single-Byte command set used in FICON™¹ devices

3.2.2

SAS

Serial Attached SCSI

3.2.3

SATA

Serial ATA

¹.FICON™ is an example of a suitable product available commercially. This information is given for the convenience of users of this standard and does not constitute an endorsement of this product by SNIA or any standards organization.

Terms and Definitions

4 Profile Introduction

4.1 Profile Overview

A profile is a specification that defines the CIM model and associated behavior for an autonomous and self-contained management domain. The CIM model includes the CIM Classes, Associations, Indications, Methods and Properties. The management domain is a set of related management tasks. A profile is uniquely identified by the name, organization and version.

In SMI-S, a profile describes the management interfaces for a class of storage subsystem, typically realized as a hardware or software product. For example, SMI-S includes profiles for arrays, FC-Switches, and logical volume manager software. The boundaries chosen for SMI-S profiles are often those of storage products, but some vendors may package things differently. For example, one vendor may choose to package an Array and an FC Switch into a single product; this can be handled in SMI-S by implementing the Array and FC Switch Profiles for this product.

A profile may add restrictions to usage and behavior, but cannot change CIM defined characteristics. For example, if a property is required in the CIM model, then it is required in a profile. On the other hand, a profile may specify that a property is required even if it is not required by the general CIM model.

In SMI-S, profiles serve several purposes:

- Specification organization - the SMI-S object model (see *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 6 Object Model General Information*) is presented as a set of profiles, each describing a type of storage element or behavior,
- Certification - SMI-S profiles form the basis for CTP certification,
- Discovery- profiles are registered with the CIM Server and advertised to clients as part of the CIM model and using SLP (see *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 9 Service Discovery*). An SMI-S client uses SLP to determine which CIM Servers host profiles it wishes to manage, then uses the CIM model to discover the actual configurations and capabilities.

A component profile is a profile that specifies a subset of a management domain. A component profile's CIM elements are scoped within a containing profile. Multiple profiles may use the same component profile. A component profile is uniquely identified by the name, organization and version.

A profile specification may include a list of the component profiles it uses. The included component profiles may be optional or mandatory by the scoping profile. The behavior of a profile is specified in this profile and its included component profiles.

For example, target devices such as RAID arrays and tape libraries may support Fibre Channel or parallel SCSI connectivity. SMI-S includes an FC Target Port Profile and a Parallel SCSI Target Port Profile that may optionally be supported by profiles representing target devices. The elements defined in the port component profiles are scoped to the ComputerSystem in the profile. For example, each LogicalPort subclass has a SystemDevice association to the profile's ComputerSystem.

In addition to sharing the purposes of profiles (as described in this section), component profiles have these purposes:

- Optional behavior - a profile may allow, but not require, an implementation to support a component profile. Although a component profile does not describe a full product, a component profile should describe an aspect of a product that is recognizable to an knowledgeable end-user such as a storage administrator,
- Reuse of functionality - some storage management behavior is common across different types of storage elements. For example, block virtualization is managed similarly in RAID arrays and logical volume managers. These common sets of functionality are specified as profiles that are shared by several other profiles.

- Decomposition - certain functionality may not be reused multiple places, but is complicated enough to document as a separate profile. For example, Disk Partition management is only used in the Host Discovered Resources profiles, but is complicated enough that it has been documented as a separate profile.

4.1.1 Terminology

A profile collects included component profiles and provides the filler needed to define the management interfaces of a particular type of subsystem. Profiles are separated into two groups. *Storage profiles* define the management interfaces for storage subsystems such as arrays or FC switches. *Generic profiles* define management interfaces for generic systems that are related to storage management. Storage and generic profiles are specified the same way in SMI-S, but generic profiles are not certified as free-standing entities, only as a dependency of a storage profile.

A *Package* is a profile that whose implementation is mandatory to comply with the requirements of all of its containing top-level profiles. Since a package is always mandatory, it is not registered with the CIM Server. Packages provide decomposition in the specification.

Profiles may be related by *specialization* - where several profiles (or component profiles) share many common elements, but are specialized for specific implementations. The SMI-S Security profiles are an example; the specializations (Authorization Profile, Security Resource Ownership Profile,...) share some classes and behavior. Profile specialization is only an artifact of the specification. It saves the reader from reading common aspects in multiple places and help the specification stay consistent across the specialized profiles. There is no information in the CIM model about the relationship between generic and specialized profiles.

4.2 Format for Profile Specifications

For each profile there is a set of information that is provided to specify the characteristics and requirements of the profile. Component profiles are also defined using this format, but they are clearly identified as component profiles.

Each profile or component profile is defined in subsections that are described in Table 1.

NOTE CIM schema diagrams are logically part of a profile description. However, they can be rather involved and cannot be easily depicted in a single diagram. As a result, the reader is advised to refer to DMTF characterizations of CIM schema diagrams.

Table 1 - Profile Components (Sheet 1 of 2)

Profile Element	Goal
Description	<p>This section provides a description of the profile and model including an overview of the objectives and functionality.</p> <p><i>Synopsis</i> describes the name, version, scoping and central classes, as well as lists the related profiles.</p> <p><i>Functionality</i> is described in a bullet-form in this section that includes functionality provided by the component profiles referenced by the profile. If a function is provided by a component profile, this is indicated, including whether the component profile is optional or required. Functionality listed in the profile is organized by Levels, and within each Level by FCAPS category, as defined in the SMI-S functionality matrix section <link>.</p> <p><i>Instance Diagrams</i>: One or more instance diagrams to highlight common implementations that employ this section of the Object Model. Instance diagrams also contain classes and associations but represent a particular configuration; multiple instances of an object may be depicted in an instance diagram.</p> <p>Finally, this section may include supporting text for properties, and methods as needed.</p>
Health & Fault Management Considerations	<p>If a profile provides optional Health & Fault Management capabilities, then this section describes the specifics of these capabilities, including:</p> <ul style="list-style-type: none"> • A table of the classes that report health information • Tables of possible states of the OperationalStatus and HealthState attributes and descriptions for those elements that report state. • Cause and Effect associations. • Standard Errors produced (including Alert Indications, Errors, CIM Errors, and Health Related Live Cycle Events).
Cascading Considerations	<p>A Profile may be a cascading profile. A cascading profile is any Profile that supports the Cascading Profile as either a mandatory or recommended component profile. If the profile is a cascading profile, this section documents cascading considerations in each of the following areas:</p> <ul style="list-style-type: none"> • Cascaded Resources – Defines the type of resources in the Cascading Profile that are associated to what type of resources in the Leaf Profile and the association. • Ownership Privileges – Identifies the Resource Control Privileges (on leaf resources) that are established by the Cascading Profile. • Limitations on Cascading Profile – Identifies any limitations on the Cascading Profile that are imposed by the Cascading in effect
Methods of the Profile	<p>This section documents the methods used in this profile.</p>
Use Cases	<p>This section documents steps required to accomplish particular tasks.</p> <p>No recipes are included in this version of the standard.</p> <p>The actual capabilities properties are documented in “Classes Used in the Profile” in this table.</p>

Table 1 - Profile Components (Sheet 2 of 2)

Profile Element	Goal
<p>CIM Elements</p>	<p>A table listing the classes, associations, and indication filters that this profile supports, and a brief description of each. Everything listed in this section is mandatory for the profile. This section shall not list optional elements.</p> <p>Prior to SMI-S 1.1.0, CIM did not have standard language for indication filters; SMI-S 1.0.x used the proposed WQL query language. This version of SMI-S uses the CQL standard query language.</p>
<p>Classes Used in the Profile</p>	<p>This section provides one table per class and lists each required and recommended property. For each required or recommended property a brief description on what information is to be encoded is identified.</p> <p>The class tables include a "Flags" column. This can contain "C" (the property is a correlatable name or a format for a name), "D" (the property is a durable name), "M" (the property is modifiable), or "N" (null is a valid value).</p>

EXPERIMENTAL

5 Generic Target Ports Profile

5.1 Synopsis

Profile Name: Generic Target Ports (Component Profile)

Version: 1.4.0

Organization: SNIA

Central Class: CIM_LogicalPort

Scoping Class: a CIM_System in a separate autonomous profile

Related Profiles: Not defined in this standard.

The Generic Target Port Profile models the generic behavior of target ports in storage systems such as disk arrays and tape libraries.

This abstract profile specification shall not be directly implemented; implementations shall be based on a profile specification that specializes the requirements of this profile.

5.2 Description

The Generic Target Port Profile models the generic behavior of target ports in storage systems such as disk arrays and tape libraries. Separate profiles specialize the Generic Target Port Profile for Fibre Channel, iSCSI, and other transports. The primary classes of the Generic Target Port Profile are LogicalPort and ProtocolEndpoint, as shown in Figure 5. Instances of subclasses of a LogicalPort (e.g., FCPort, EthernetPort) represent the logical aspects of ports, independent from command protocols (such as SCSI). Instances of subclasses of ProtocolEndpoint (e.g., SCSIProtocolEndpoint or ATA ProtocolEndpoint) represent command protocols in use on the port.

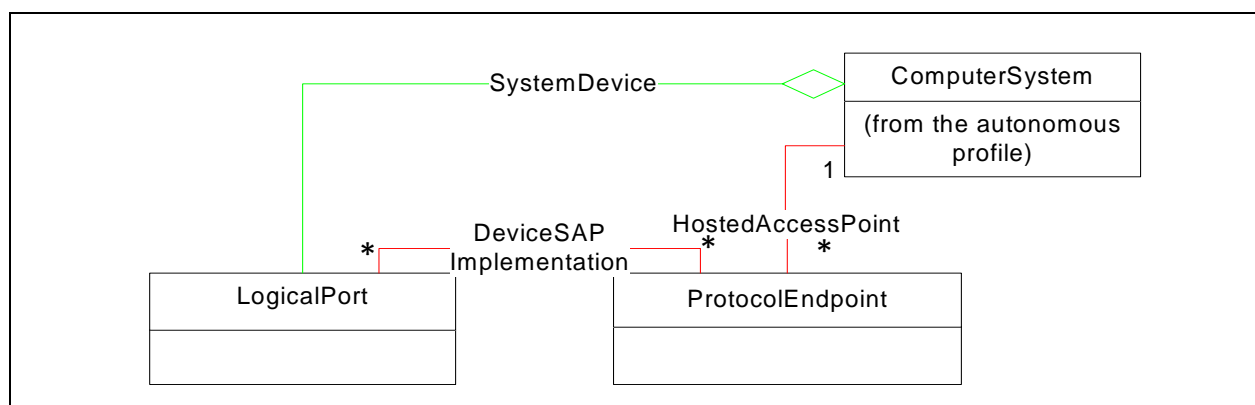


Figure 5 - Generic Target Port Classes

5.3 Implementation

Subclasses of ProtocolEndpoint represent command protocols supported by the port. SCSIProtocolEndpoint represents SCSI as a protocol, independent of specific transports or device types – i.e., the behavior described in the SCSI Primary Commands (SPC) and SCSI Architecture Model (SAM) specifications from T10. SCSIProtocolEndpoint.Role indicates whether this protocol endpoint instance represents a SCSI Target or target. For target port profiles, Role shall be “Target” or “Both Initiator and Target”. iSCSIProtocolEndpoint specializes SCSIProtocolEndpoint with additional iSCSI-specific properties.

ATAProtocolEndpoint represents the ATA command protocol. SBPProtocolEndpoint represents Single Byte protocol used with mainframes. ProtocolEndpoint is associated to a System instance with Hosted Access Point.

LogicalPort subclasses specify the type of transport. If the port is subclassed directly from LogicalPort it indicates it is connected to a bus. If the port is further subclassed from NetworkPort it indicates the port is capable of being used in a network. Specializations of this profile shall specify the appropriate subclass of LogicalPort. Figure 6 shows the subclasses of LogicalPort.

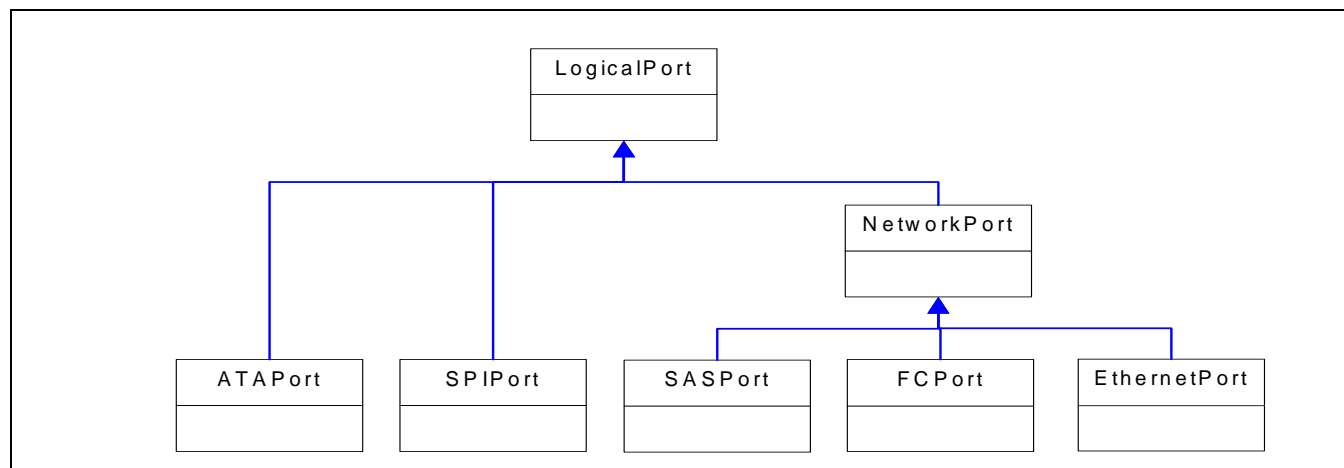


Figure 6 - LogicalPort Class Hierarchy

A property on LogicalPort called UsageRestriction is indicates whether the port is restricted to use as a “front end” (target) or a “back end” (Target) interface or both. Note that port may not have a restriction and the actual point-in-time role is modeled in SCISProtocolEndpoint.Role. SystemDevice associates LogicalPort to a System.

ProtocolEndpoint and LogicalPort are associated with DeviceSAPImplementation. For most transports, the command protocol is implemented in the port hardware and there is 1-1 cardinality between the LogicalPort and ProtocolEndpoint instances. iSCSI is an exception, many-to-many relationships are possible between EthernetPort and iSCSIProtocolEndpoint instances

ProtocolController (in the Mapping and Masking Profile) represents the SCSI (or SB) ‘view’ of ports and logical devices seen by target systems (e.g., arrays). In a system supporting Mapping and Masking, zero or more views exist; defined by the customer to expose subsets of logical units to certain Targets. SAPAvailableForElement connects ProtocolEndpoint from a target ports profile to SCISProtocolController instances from the Mapping/Masking Profile. iSCSI and SB have protocol-specific, secondary uses of ProtocolController.

Figure 7 depicts a generic storage device with elements from a target ports profile, the Mapping/Masking Profile, and a target device profile. The LogicalDevice object represents logical units that are visible to

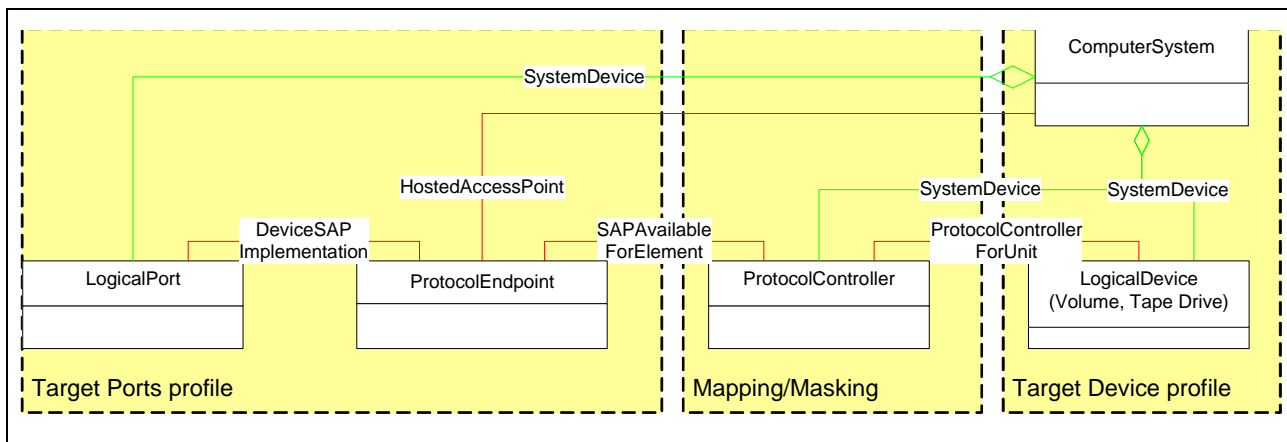


Figure 7 - Generic Target with LUN Masking

external systems. It is subclassed to StorageVolume, TapeDrive, etc. to identify the device type.

5.3.1 Modeling SCSI/SB Logical Units

The SCSI standard inquiry response includes a Device Type property with integers representing types of devices. Most of these devices types have a CIM analog. Devices that are used primarily for management are modeled as SCSIArbitraryLogicalUnit. SCSIArbitraryLogicalUnit.DeviceType maps to SCSI device types. Table 2 describes how common storage devices are modeled in CIM.

Table 2 - Modeling of Common Storage Devices in CIM

SCSI Device Type	Inquiry Device Type	LogicalDevice subclass
DirectAccessDevice	0	DiskDrive or StorageVolume
SequentialAccessDevice	1	TapeDrive
WriteOnceDevice	4	WormDrive
CD-ROM	5	CDROMDrive
MediaChanger	8	MediaTransferDevice
ArrayController	0xc	SCSIArbitraryLogicalUnit DeviceType="SCSI SCC Device"
SES	0xd	SCSIArbitraryLogicalUnit DeviceType="SCSI SES"
Other		SCSIArbitraryLogicalUnit DeviceType="Other"
Unknown		SCSIArbitraryLogicalUnit DeviceType="Unknown"
DirectAccessDevice	0	DiskDrive or StorageVolume

All devices (logical units) visible to external systems shall be modeled.

5.4 Methods of the Profile

5.4.1 Extrinsic Methods

Not defined in this standard.

5.4.2 Intrinsic Methods

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

5.5 Use Cases

Not defined in this standard.

5.6 CIM Elements

Table 3 describes the CIM elements for Generic Target Ports.

Table 3 - CIM Elements for Generic Target Ports

Element Name	Requirement	Description
5.6.1 CIM_DeviceSAPImplementation	Mandatory	Associates front-end LogicalPort and target ProtocolEndpoint.
5.6.2 CIM_HostedAccessPoint	Mandatory	Associates ComputerSystem to ProtocolEndpoint.
5.6.3 CIM_LogicalPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
5.6.4 CIM_ProtocolEndpoint	Mandatory	ProtocolEndpoint representing support for SCSI, ATA, or SB command set.
5.6.5 CIM_SystemDevice (Port)	Mandatory	Associates ComputerSystem to LogicalPort.

5.6.1 CIM_DeviceSAPImplementation

Associates front-end LogicalPort and target ProtocolEndpoint.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 4 describes class CIM_DeviceSAPImplementation.

Table 4 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSI, ATA, or SB ProtocolEndpoint.
Antecedent		Mandatory	Reference to Port.

5.6.2 CIM_HostedAccessPoint

Associates ComputerSystem to ProtocolEndpoint. Limit to targets.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 5 describes class CIM_HostedAccessPoint.

Table 5 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in the referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint, ATAProtocolEndpoint or SBProtocolEndpoint.

5.6.3 CIM_LogicalPort

Represents the logical aspects of the physical port and may have multiple associated protocols.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 6 describes class CIM_LogicalPort.

Table 6 - SMI Referenced Properties/Methods for CIM_LogicalPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 2 for ports restricted to Front-end only or 4 if the port is unrestricted.
PortType		Mandatory	VALUE and DESC should be set appropriately for each specialized target port profile.

5.6.4 CIM_ProtocolEndpoint

ProtocolEndpoint representing support for SCSI, ATA, or SB command set. Shall be subclassed in specialized profiles.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 7 describes class CIM_ProtocolEndpoint.

Table 7 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocollFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI','ATA', or 'SB'.

5.6.5 CIM_SystemDevice (Port)

Associates ComputerSystem to LogicalPort.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 8 describes class CIM_SystemDevice (Port).

Table 8 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem in the referencing profile.
PartComponent		Mandatory	Reference to Port.

EXPERIMENTAL

STABLE

6 FC Target Ports Profile

6.1 Synopsis

Profile Name: FC Target Ports (Component Profile)

Version: 1.7.0

Organization: SNIA

Central Class: CIM_FCPort

Scoping Class: a CIM_ComputerSystem in a referencing autonomous profile

Related Profiles: Table 9 describes the related profiles for FC Target Ports.

Table 9 - Related Profiles for FC Target Ports

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

6.2 Description

The FC Target Port Profile models the Fibre Channel specific aspects of a target storage system.

6.3 Implementation

For Fibre Channel ports, the concrete subclass of LogicalPort is FCPort. FCPort is always associated 1-1 with a SCSIProtocolEndpoint instance.

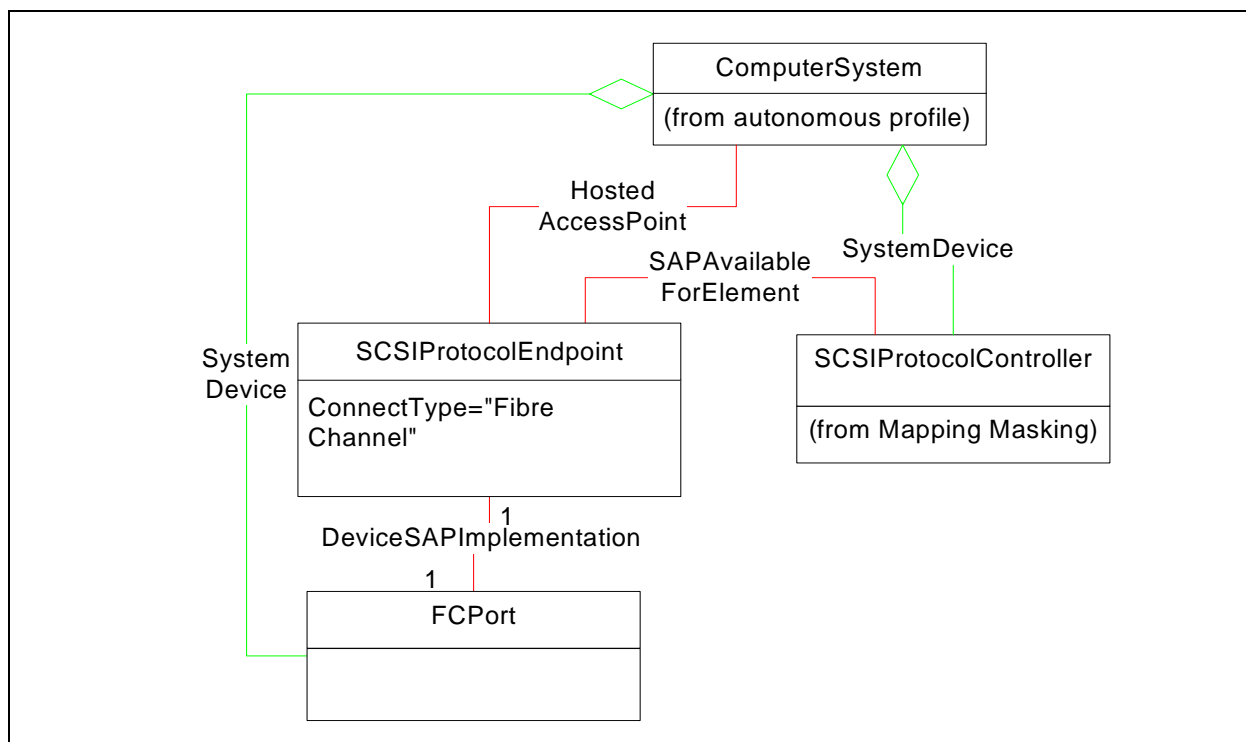


Figure 8 - FC Target Port Instance Diagram

6.4 Durable Names and Correlatable IDs of the Profile

FCPort.PermanantAddress shall contain the port's Port WWN.

6.5 Health and Fault Management

Table 10 describes FCPort OperationalStatus.

Table 10 - FCPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

6.6 Supported Profiles and Packages

Not defined in this standard.

6.7 Extrinsic Methods of this Profile

Not defined in this standard.

6.8 Client Considerations and Recipes

No recipes are defined in this version of the standard.

6.9 CIM Elements

Table 11 describes the CIM elements for FC Target Ports.

Table 11 - CIM Elements for FC Target Ports

Element Name	Requirement	Description
6.9.1 CIM_DeviceSAPImplementation	Mandatory	Associates FCPort and SCSIProtocolEndpoint.
6.9.2 CIM_FCPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
6.9.3 CIM_HostedAccessPoint	Mandatory	Associates ComputerSystem to SCSIProtocolEndpoint.
6.9.4 CIM_SCSIProtocolEndpoint	Mandatory	Represents management characteristics related to the SCSI command set.
6.9.5 CIM_SystemDevice (Port)	Mandatory	Associates controller ComputerSystem to FCPort.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_FCPort	Mandatory	Create FCPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::OperationalStatus <> PreviousInstance.CIM_FCPort::OperationalStatus	Mandatory	CQL -Change to FCPort OperationalStatus.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::Speed <> PreviousInstance.CIM_FCPort::Speed	Optional	CQL -Change to FCPort Speed.

Table 11 - CIM Elements for FC Target Ports

Element Name	Requirement	Description
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::NetworkAddresses <> PreviousInstance.CIM_FCPort::NetworkAddresses	Optional	CQL -Change to FCPort NetworkAddresses.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FCPort	Mandatory	Delete FCPort.

6.9.1 CIM_DeviceSAPImplementation

Associates FCPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 12 describes class CIM_DeviceSAPImplementation.

Table 12 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.
Antecedent		Mandatory	Reference to FCPort.

6.9.2 CIM_FCPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 13 describes class CIM_FCPort.

Table 13 - SMI Referenced Properties/Methods for CIM_FCPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 2 for ports restricted to Front-end only or 4 if the port is unrestricted.

Table 13 - SMI Referenced Properties/Methods for CIM_FCPort

Properties	Flags	Requirement	Description & Notes
PortType		Mandatory	Shall be 0 1 10 11 12 13 14 15 16 17 18 (Unknown or Other or N or NL or F/NL or Nx or E or F or FL or B or G).
OtherPortType		Conditional	Required when PortType is Other
PermanentAddress	CD	Mandatory	Port WWN. Shall be 16 unseparated uppercase hex digits.
Speed		Optional	Speed in bits per second. Shall be 0, 1062500000 (1GFC), 2125000000 (2GFC), 4250000000 (4GFC), 8500000000 (8GFC), 10518750000 (10GFC), 14025000000 (16GFC), 21037500000 (20GFC) or 28500000000 (32GFC).
MaxSpeed		Optional	Maximum Port Speed.
NetworkAddresses		Optional	For Fibre Channel end device ports, the Fibre Channel ID. Shall be 8 unseparated upper case hex digits.

6.9.3 CIM_HostedAccessPoint

Associates ComputerSystem to SCSIProtocolEndpoint. Limit to targets. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 14 describes class CIM_HostedAccessPoint.

Table 14 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.

6.9.4 CIM_SCSIProtocolEndpoint

Represents management characteristics related to the SCSI command set. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 15 describes class CIM_SCSIProtocolEndpoint.

Table 15 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).
Role		Mandatory	Shall be 3 (Target) or 4 (Both Initiator and Target).

6.9.5 CIM_SystemDevice (Port)

Associates controller ComputerSystem to FCPort. The class definition specializes the CIM_SystemDevice definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 16 describes class CIM_SystemDevice (Port).

Table 16 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem in the referencing profile.
PartComponent		Mandatory	Reference to FCPort.

STABLE

EXPERIMENTAL

7 FCoE Target Ports Profile

7.1 Synopsis

Profile Name: FCoE Target Ports (Component Profile)

Version: 1.7.0

Organization: SNIA

Central Class: FCPort

Scoping Class: a ComputerSystem in a referencing autonomous profile

Related Profiles: Table 17 describes the related profiles for FCoE Target Ports.

Table 17 - Related Profiles for FCoE Target Ports

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

7.2 Description

The FCoE Target Ports Profile is a component profile that models the behavior of the Fibre Channel over Ethernet (FCoE) functionality.

Figure 9 shows the topology of FCoE where the Fibre Channel protocol is carried over the Ethernet.

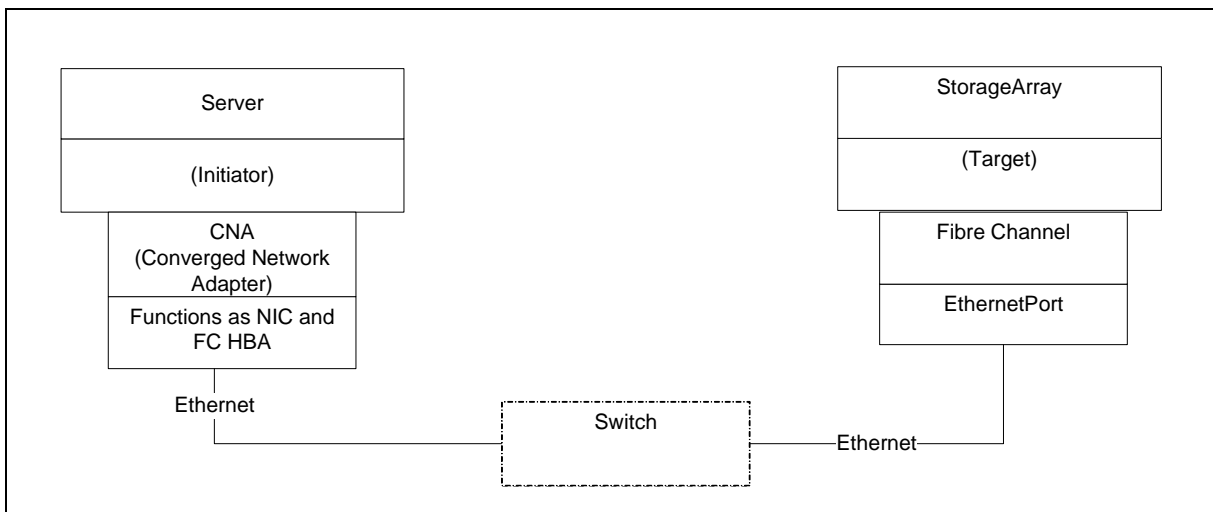


Figure 9 - FCoE Topology

7.3 Implementation

The FCPort is always associated 1-1 with a SCSIProtocolEndpoint instance. The EthernetPort is associated to 0 or more FCPorts. The EthernetPort used for FCoE protocol, shown in Figure 10, has a PortDiscriminator property with the value of "SNIA:FCoE".

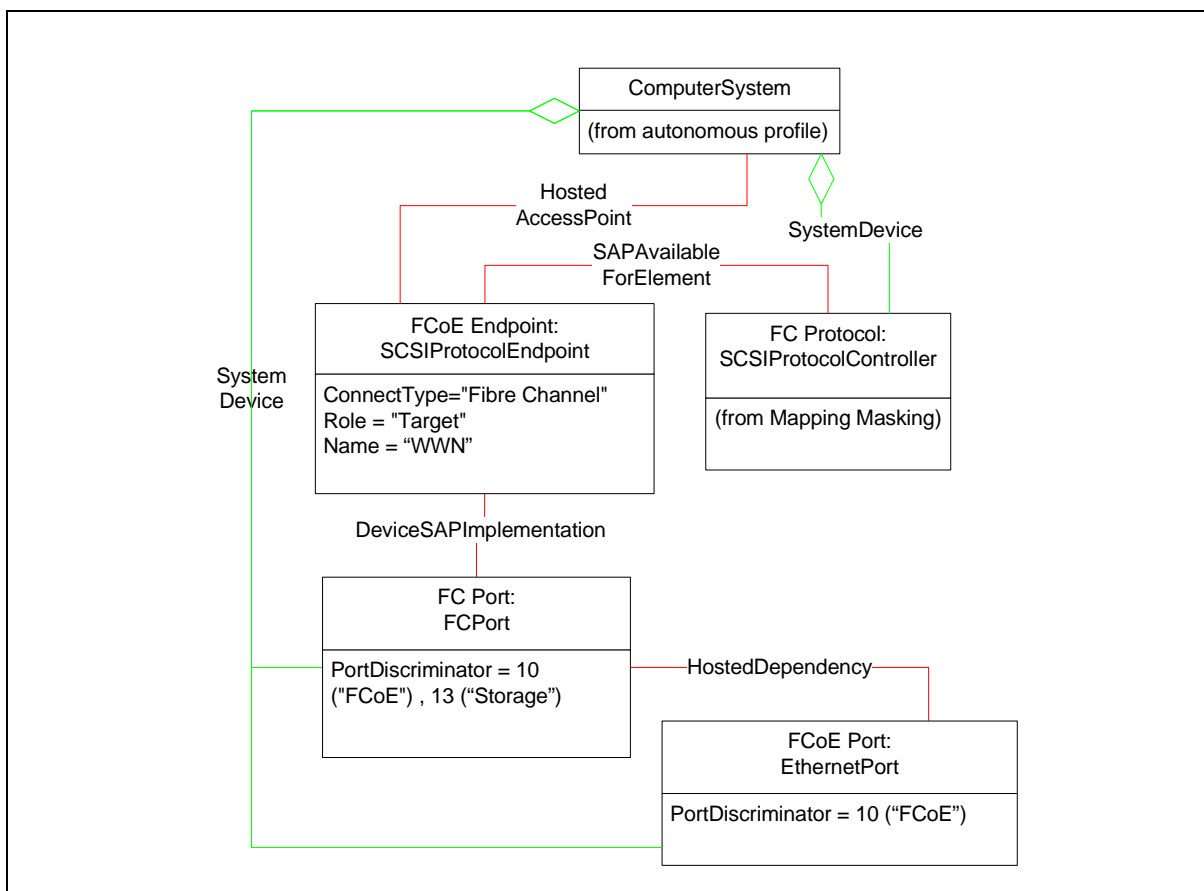


Figure 10 - EthernetPort used for FCoE

7.4 Durable Names and Correlatable IDs of the Profile

FCPort.PermanantAddress shall contain the port's Port WWN.

7.4.1 Health and Fault Management Consideration

Table 18 describes FCPort OperationalStatus.

Table 18 - FCPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

7.4.2 Cascading Considerations

Not defined in this standard.

7.5 Methods

Not defined in this standard.

7.6 Use Cases

Not defined in this standard.

7.7 CIM Elements

Table 19 describes the CIM elements for FCoE Target Ports.

Table 19 - CIM Elements for FCoE Target Ports

Element Name	Requirement	Description
7.7.1 CIM_DeviceSAPImplementation	Mandatory	Associates FCPort and SCSIProtocolEndpoint.
7.7.2 CIM_EthernetPort	Mandatory	
7.7.3 CIM_FCPort (For FCoE)	Mandatory	FCPort specialized to handle FCoE.
7.7.4 CIM_HostedAccessPoint	Mandatory	Associates ComputerSystem to SCSIProtocolEndpoint.
7.7.5 CIM_HostedDependency (NetworkPort to FCPort)	Mandatory	Association between EthernetPort and FCPort.
7.7.6 CIM_LogicalPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
7.7.7 CIM_SCSIProtocolEndpoint	Mandatory	Represents management characteristics related to the SCSI command set.
7.7.8 CIM_SystemDevice (Port)	Mandatory	Associates controller ComputerSystem to FCPort.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_FCPort	Mandatory	Create FCPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::OperationalStatus <> PreviousInstance.CIM_FCPort::OperationalStatus	Mandatory	CQL -Change to FCPort OperationalStatus.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::Speed <> PreviousInstance.CIM_FCPort::Speed	Optional	CQL -Change to FCPort Speed.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::NetworkAddresses <> PreviousInstance.CIM_FCPort::NetworkAddresses	Optional	CQL -Change to FCPort NetworkAddresses.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FCPort	Mandatory	Delete FCPort.

7.7.1 CIM_DeviceSAPImplementation

Associates FCPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 20 describes class CIM_DeviceSAPImplementation.

Table 20 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.
Antecedent		Mandatory	Reference to FCPort.

7.7.2 CIM_EthernetPort

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 21 describes class CIM_EthernetPort.

Table 21 - SMI Referenced Properties/Methods for CIM_EthernetPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
LinkTechnology		Mandatory	Shall be 2 (Ethernet).
OperationalStatus		Mandatory	Shall be 0 (Unknown), 2 (OK), 6 (Error), 10 (Stopped), or 11 (In Service).
PermanentAddress	CD	Mandatory	The MAC Address. Shall be formatted as 12 un-separated upper case hex digits.
PortDiscriminator		Mandatory	Discriminates the supported context of this EthernetPort. Value: 10 ('FCoE').

7.7.3 CIM_FCPort (For FCoE)

FCPort specialized to handle FCoE.

Requirement: Mandatory

Table 22 describes class CIM_FCPort (For FCoE).

Table 22 - SMI Referenced Properties/Methods for CIM_FCPort (For FCoE)

Properties	Flags	Requirement	Description & Notes
PortType		Mandatory	Shall be 0 1 10 11 12 13 14 15 16 17 18 (Unknown or Other or N or NL or F/NL or Nx or E or F or FL or B or G).
PermanentAddress	CD	Mandatory	Port WWN. Shall be 16 unseparated uppercase hex digits.
SupportedCOS		Optional	

Table 22 - SMI Referenced Properties/Methods for CIM_FCPort (For FCoE)

Properties	Flags	Requirement	Description & Notes
ActiveCOS		Optional	
SupportedFC4Types		Optional	
ActiveFC4Types		Optional	
Speed		Optional	Speed in bits per second. Shall be 0, 1062500000 (1GFC), 2125000000 (2GFC), 4250000000 (4GFC), 8500000000 (8GFC), 10518750000 (10GFC), 14025000000 (16GFC), 21037500000 (20GFC) or 28500000000 (32GFC).
MaxSpeed		Optional	Maximum Port Speed.
NetworkAddresses		Optional	For Fibre Channel end device ports, the Fibre Channel ID. Shall be 16 un-separated upper case hex digits.
PortDiscriminator		Mandatory	Discriminates the supported context of this FCPort. Value: 10, 13 ('FCoE', 'Storage').

7.7.4 CIM_HostedAccessPoint

Associates ComputerSystem to SCSIProtocolEndpoint. Limit to targets. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 23 describes class CIM_HostedAccessPoint.

Table 23 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.

7.7.5 CIM_HostedDependency (NetworkPort to FCPort)

Association between EthernetPort and FCPort.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 24 describes class CIM_HostedDependency (NetworkPort to FCPort).

Table 24 - SMI Referenced Properties/Methods for CIM_HostedDependency (NetworkPort to FCPort)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to EthernetPort.
Dependent		Mandatory	Reference to FCPort.

7.7.6 CIM_LogicalPort

Represents the logical aspects of the physical port and may have multiple associated protocols.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 25 describes class CIM_LogicalPort.

Table 25 - SMI Referenced Properties/Methods for CIM_LogicalPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 2 for ports restricted to Front-end only or 4 if the port is unrestricted.
PortType		Mandatory	VALUE and DESC should be set appropriately for each specialized target port profile.

7.7.7 CIM_SCSIProtocolEndpoint

Represents management characteristics related to the SCSI command set. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 26 describes class CIM_SCSIProtocolEndpoint.

Table 26 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).
Role		Mandatory	Shall be 3 (Target) or 4 (Both Initiator and Target).

7.7.8 CIM_SystemDevice (Port)

Associates controller ComputerSystem to FCPort. The class definition specializes the CIM_SystemDevice definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 27 describes class CIM_SystemDevice (Port).

Table 27 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem in the referencing profile.
PartComponent		Mandatory	Reference to FCPort.

EXPERIMENTAL

FCoE Target Ports Profile

STABLE**8 iSCSI Target Ports Profile****8.1 Synopsis**

Profile Name: iSCSI Target Ports (Component Profile)

Version: 1.6.0

Organization: SNIA

Central Class: EthernetPort

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

Models an iSCSI target port

8.2 Description

The iSCSI Target Ports Profile describes the iSCSI specific aspects of a target device.

8.3 Implementation

iSCSI terminology is different than that used in other parts of SMI-S. Table 28 provides a map of terminology from iSCSI standards and CIM class names used in this standard. iSCSI does have a specific naming requirement for SCSIProtocolController that is described in Table 28.

Table 28 - iSCSI Terminology and SMI-S Class Names

iSCSI Term	CIM Class Name	Notes
Network Entity	ComputerSystem	The Network Entity represents a device or gateway that is accessible from the IP network. A Network Entity shall have one or more Network Portals, each of which can be used to gain access to the IP network by some iSCSI Nodes contained in that Network Entity.
Session	iSCSISession	The group of TCP connections that link a Target with a target form a session (loosely equivalent to a SCSI I-T nexus). TCP connections can be added and removed from a session. Across all connections within a session, a Target sees one and the same target.
Connection	NetworkPipe	A connection is a TCP connection. Communication between the Target and target occurs over one or more TCP connections. The TCP connections carry control messages, SCSI commands, parameters, and data within iSCSI Protocol Data Units (iSCSI PDUs).
SCSI Port	iSCSIProtocolEndpoint	A SCSI Port using an iSCSI service delivery subsystem. A collection of Network Portals that together act as a SCSI Target or target.
Portal Group	SystemSpecificCollection	iSCSI supports multiple connections within the same session; some implementations will have the ability to combine connections in a session across multiple Network Portals. A Portal Group defines a set of Network Portals within an iSCSI Network Entity that collectively supports the capability of coordinating a session with connections spanning these portals. Not all Network Portals within a Portal Group need participate in every session connected through that Portal Group. One or more Portal Groups may provide access to an iSCSI Node. Each Network Portal, as utilized by a given iSCSI Node, belongs to exactly one portal group within that node.

Table 28 - iSCSI Terminology and SMI-S Class Names

Network Portal	TCPProtocolEndpoint, IPProtocolEndpoint, EthernetPort	The Network Portal is a component of a Network Entity that has a TCP/IP network address and that may be used by an iSCSI Node within that Network Entity for the connection(s) within one of its iSCSI sessions. A Network Portal in a Target is identified by its IP address. A Network Portal in a target is identified by its IP address and its listening TCP port.
Node	SCSIProtocolController	The iSCSI Node represents a single iSCSI Target or iSCSI target. There are one or more iSCSI Nodes within a Network Entity. The iSCSI Node is accessible via one or more Network Portals. An iSCSI Node is identified by its iSCSI Name. The separation of the iSCSI Name from the addresses used by and for the iSCSI Node allows multiple iSCSI nodes to use the same address, and the same iSCSI node to use multiple addresses.

Figure 11: "iSCSI Target Ports Profile Instance Diagram" is a class diagram for iSCSI Target Ports and uses the UML instance naming notation (InstanceName:ClassName) with the iSCSI-style names before the CIM names. Figure 28: "iSCSI Terminology and SMI-S Class Names" explains the use of all these objects.

Note that ComputerSystem, SCSIProtocolController and StorageVolume are not actually part of this profile; they would be the parts of the Array Profile that associate with the iSCSI-specific classes.

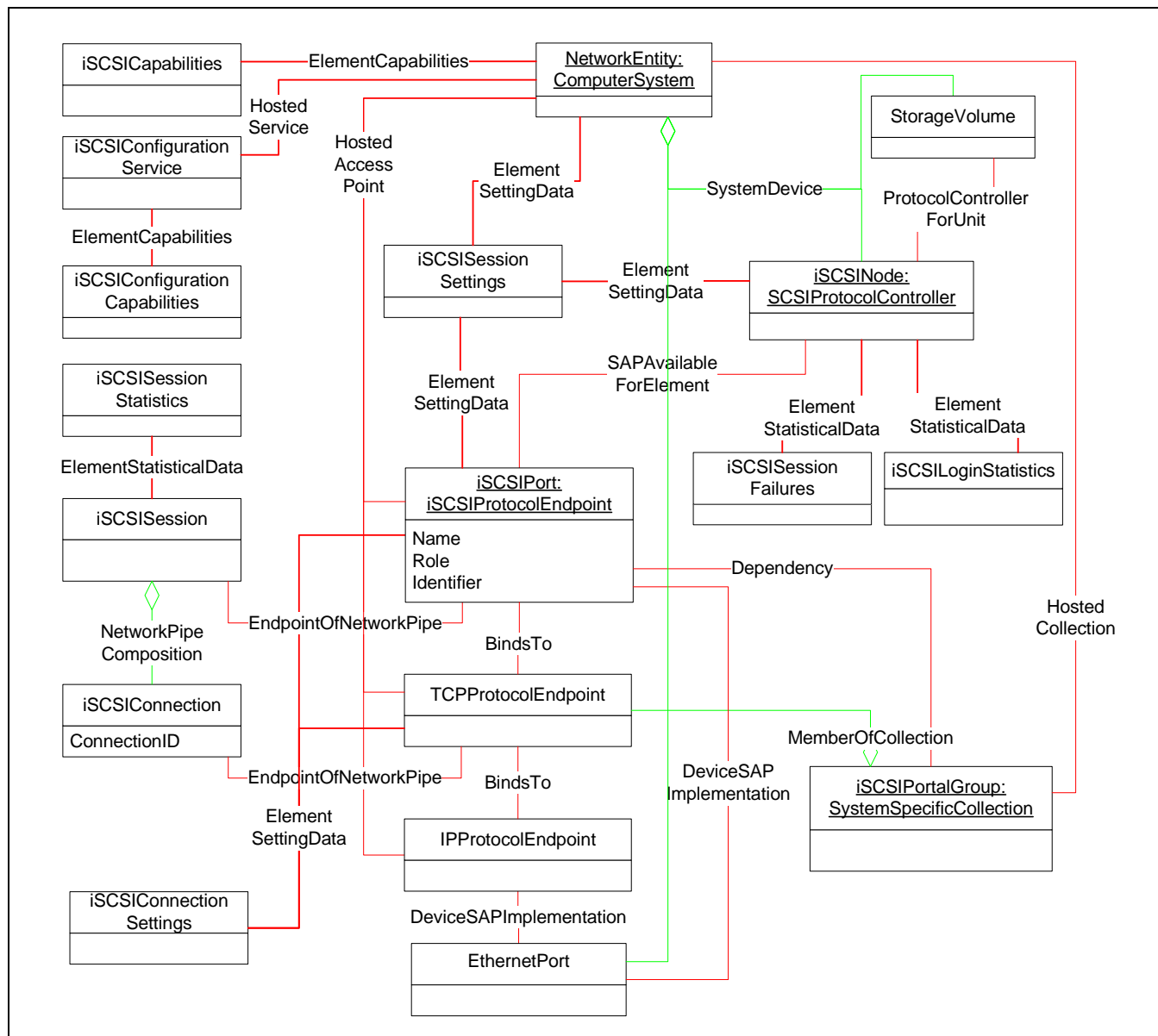


Figure 11 - iSCSI Target Ports Profile Instance Diagram

8.3.1 Mapping and Masking Considerations

The class SCSIProtocolController is used in the Mapping and Masking Profile to model a “view”, which is a set of logical devices exposed to a Target. It is in a sense a virtual SCSI device, but carries no SCSI device name when used with the other Target Ports profiles such as the FC Target Port Profile. In fact the class is even not part of these component profiles.

The iSCSI Target Ports Profile however uses SCSIProtocolController to model the iSCSI Node which is the SCSI Device as defined in the SAM specification. It has a SCSI device name which is the iSCSI Node Name. Thus the presence of instances of SCSIProtocolController with this profile has multiple meanings. Whereas there may be no instances of SCSIProtocolController with other Target Port component profiles until created as views by the Mapping and Masking method ExposePaths, instances of

SCSIProtocolControllers as iSCSINodes can be brought into existence by the iSCSI method CreateiSCSINode. The instances can then be used as inputs to ExposePaths to grant access by Targets to logical devices through the Node. This initial SCSIProtocolController that was created as a Node will be the first view. Additional “view” ProtocolControllers created by ExposePaths would carry the same iSCSI Node name to convey that they represent the same underlying Node.

8.3.2 Settings

An iSCSI Session is established between a Target Port and a Target Port through the establishment of an initial iSCSI Connection, which happens during the “Leading” Login. At this time the operational properties for the Session are negotiated and also the operational properties for the initial Connection. Additional Connections for the Session are established through subsequent logins. For many operational properties both the Target and Target have settings that specify the starting position for the negotiation process. The settings for negotiating Session-wide operational properties (found in iSCSISession) are in iSCSISessionSettings. Likewise the settings for negotiating Connection level operational properties (found in iSCSI Connection) are in iSCSIConnectionSettings. For example, iSCSISessionSettings contains the property MaxConnectionsPerSession, which is the value that the local system (which in this profile is the Target) would like to use for Session. When the leading login is complete the actual value agreed upon with the Target is in the property MaxConnectionsPerSession in iSCSI Session.

Different implementations may scope the settings classes differently.

iSCSISessionSettings can be associated to any one of the following classes:

- iSCSIProtocolEndpoint: The Settings apply to Sessions created on the iSCSI Port represented by the iSCSIProtocolEndpoint.
- SCSIProtocolController: The Settings apply to Sessions created on all iSCSIProtocolEndpoint belonging to the iSCSI Node represented by the SCSIProtocolController.
- ComputerSystem: The Settings apply to Sessions created on all iSCSIProtocolEndpoints belonging to all SCSIProtocolControllers belonging to the ComputerSystem.

iSCSIConnectionSettings can be associated to any one of the following classes:

- TCPProtocolEndpoint: The Settings apply to each Connection created using the Network Portal represented by the TCPProtocolEndpoint, regardless of which iSCSIProtocolEndpoint owns the Session that the Connection belongs to.
- iSCSIProtocolEndpoint: The Settings apply to Connections using NetworkPortals to which the iSCSIProtocolEndpoint is bound and belonging to Sessions on that same iSCSIProtocolEndpoint.

EXPERIMENTAL

NOTE The support on iSCSI Session is conditional, which means it is only supported when ‘iSCSI Session’ is included in iSCSICapabilities.SupportedFeatures.

EXPERIMENTAL

8.3.3 Durable Names and Correlatable IDs of the Profile

The Name property for the iSCSI node (SCSIProtocolController) shall be a compliant iSCSI name as described in *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5, 7.8 "iSCSI Names"*. NameFormat shall be set to “iSCSI Name”.

The Name property for iSCSIProtocolEndpoint shall be a compliant iSCSI name as described in *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5, 7.8 "iSCSI Names"* ConnectionType shall be set to "iSCSI".

8.4 Health and Fault Management

Table 29 defines the SMI-S-defined meanings of the OperationalStatus property for EthernetPort used in the SB Target Port Profile.

Table 29 - EthernetPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

8.5 Methods of this Profile

The iSCSIConfigurationService provides the following methods that allow a client to manipulate iSCSIProtocolEndpoints in an iSCSI Target Node. The class iSCSIProtocolController models the iSCSI Target Port. The instance of the service is scoped by an instance of ComputerSystem that represents that Network Entity. The capabilities of this service are defined in the companion class iSCSIConfigurationCapabilities.

8.5.1 CreateiSCSINode

This method creates an iSCSI Node in the form of an instance of SCSIProtocolController. As part of the creation process a SystemDevice association is created between the new SCSIProtocolController and the scoping Network Entity (ComputerSystem) hosting this service.

CreateiSCSINode

IN, string **Alias**,

The iSCSI Alias for the new Node.

OUT, SCSIProtocolController REF **iSCSINode**,

A reference to the new SCSIProtocolController that is created.

8.5.1.1 Return Values

Success

Not Supported

Unspecified Error

Timeout

Failed

Node Creation Not Supported

Alias in use by Other Node

8.5.1.2 Created Instances

SCSIProtocolController

SystemDevice

8.5.1.3 Deleted Instances

Not defined in this standard.

8.5.1.4 Modified Instances

Not defined in this standard.

8.5.2 DeleteiSCSINode

The method deletes an instance of SCSIProtocolController representing an iSCSI Node and all associations in which this SCSIProtocolController is referenced. If Sessions are active on iSCSIProtocolEndpoints belonging to this Node an error will be returned. If no Sessions are active the scoped iSCSIProtocolEndpoints will be deleted.

DeleteiSCSINode

IN, SCSIProtocolController REF **iSCSINode**

The SCSIProtocolController to be deleted.

8.5.2.1 Return Values

Success

Not Supported

Unspecified Error

Timeout

Failed

Invalid Parameter

SCSIProtocolController Non-existent

Sessions Active on Node Ports

8.5.2.2 Created Instances

Not defined in this standard.

8.5.2.3 Deleted Instances

SCSIProtocolController

SystemDevice

iSCSIProtocolEndpoint

HostedAccessPoint

SAPAvailableForElement

BindsTo

8.5.2.4 Modified Instances

Not defined in this standard.

8.5.3 CreateiSCSIProtocolEndpoint

This method creates an iSCSI Port in the form of an instance of iSCSIProtocolEndpoint. As part of the creation process the iSCSIProtocolEndpoint is 'bound to' the underlying TCPProtocolEndpoints which are specified as inputs by creating instances of the BindsTo association between the new instance and those instances. In addition, an instance of SAPAvailableForElement is created between the specified SCSIProtocolController and the new instance of iSCSIProtocolEndpoint.

CreateiSCSIProtocolEndpoint

IN, SCSIProtocolController REF **iSCSINode**,

The SCSIProtocolController instance representing the iSCSI Node that will contain the iSCSI Port.

IN, uint16 **Role**,

For iSCSI, each iSCSIProtocolEndpoint acts as either a target or a Target endpoint. This property indicates which role this iSCSIProtocolEndpoint implements.

IN, string **Identifier**,

The Identifier shall contain the Target Portal Group Tag (TGPT). Each iSCSIProtocolEndpoint (iSCSI port) associated to a common SCSIProtocolController (iSCSI node) has a unique Identifier. This field is a string that contains 12 hexadecimal digits. If the property IdentifierSelectionSupported in class iSCSIConfigurationCapabilities is false, this parameter shall be set to NULL.

IN, ProtocolEndpoint REF **NetworkPortals[]**,

An Array of References to TCPProtocolEndpoints representing Target Network Portals. The TCPProtocolEndpoints specified each shall be associated to an instance of IPProtocolEndpoint via a BindsTo association in order to provide the Target Network Portal functionality. The selected Portal endpoints shall be from the same SystemSpecificCollection, which represents a Portal Group.

OUT, iSCSIProtocolEndpoint REF **iSCSIPort**,

A reference to the new iSCSIProtocolEndpoint that is created.

8.5.3.1 Return Values

Success

Not Supported

Unspecified Error

Timeout

Failed

SCSIProtocolController Non-existent

Role Not Supported By Specified SCSIProtocolController

Identifier In Use, Not Unique

Identifier Selection Not Supported

ProtocolEndpoint Non-Existent

TCPProtocolEndpoint Not Bound To Underlying IPProtocolEndpoint

TCPProtocolEndpoint In Use By Other iSCSIProtocolEndpoint In Same Target SCSIProtocolController.

ProtocolEndpoints Not From Same Endpoint Collection

8.5.3.2 Created Instances

iSCSIProtocolEndpoint

HostedAccessPoint

SAPAvailableForElement

BindsTo

8.5.3.3 Deleted Instances

Not defined in this standard

8.5.3.4 Modified Instances

Not defined in this standard

8.5.4 DeleteiSCSIProtocolEndpoint

The method deletes an instance of iSCSIProtocolEndpoint and all associations in which this iSCSIProtocolEndpoint is referenced.

DeleteiSCSIProtocolEndpoint

IN, iSCSIProtocolEndpoint REF **iSCSIPort**

The iSCSIProtocolEndpoint to be deleted.

8.5.4.1 Return Values

Success

Not Supported

Unspecified Error

Timeout

Failed

Invalid Parameter

Endpoint Non-existent

8.5.4.2 Created Instances

Not defined in this standard

8.5.4.3 Deleted Instances

iSCSIProtocolEndpoint

HostedAccessPoint

SAPAvailableForElement

BindsTo

8.5.4.4 Modified Instances

Not defined in this standard

8.5.5 Bind*i*SCSIProtocolEndpoint

This method provides for modification of an existing iSCSI Port by associating a TCPProtocolEndpoint representing a Target Network Portal to the iSCSIProtocolEndpoint. The association is persisted as an instance of BindsTo. The selected Portal endpoint shall be from the same SystemSpecificCollection, which represents a Portal Group, as those endpoints currently bound to the iSCSIProtocolEndpoint.

This action is intended to be reversed by the use of the intrinsic method '**DeleteInstance**'.

Bind*i*SCSIProtocolEndPoint

IN, iSCSIProtocolEndpoint REF **iSCSIPort**,

A reference to the iSCSIProtocolEndpoint

IN, ProtocolEndpoint REF **NetworkPortal**

An instance of TCPProtocolEndpoint representing the Network Portal to be added

8.5.5.1 Return Values

Success

Not Supported

Unspecified Error

Timeout

Failed

Invalid Parameter

ProtocolEndpoint Non-Existent

TCPProtocolEndpoint Not Bound To Underlying IPProtocolEndpoint

ProtocolEndpoint In Use By Other iSCSIProtocolEndpoint In Same Target SCSIProtocolController

ProtocolEndpoint Not From Same Endpoint Collection

8.5.5.2 Created Instances

BindsTo

8.5.5.3 Deleted Instances

Not defined in this standard

8.5.5.4 Modified Instances

Not defined in this standard

8.6 Client Considerations and Recipes

No recipes are defined in this version of the standard.

8.7 CIM Elements

Table 30 describes the CIM elements for iSCSI Target Ports.

Table 30 - CIM Elements for iSCSI Target Ports

Element Name	Requirement	Description
8.7.1 CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint)	Mandatory	
8.7.2 CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)	Mandatory	
8.7.3 CIM_ConcreteDependency	Conditional	This is required if CIM_SystemSpecificCollection is used.
8.7.4 CIM_DeviceSAPImplementation (EthernetPort to IPProtocolEndpoint)	Optional	
8.7.5 CIM_DeviceSAPImplementation (EthernetPort to iSCSIProtocolEndpoint)	Optional	
8.7.6 CIM_ElementCapabilities (CIM_iSCSICapabilities to System)	Mandatory	
8.7.7 CIM_ElementCapabilities (iSCSIConfigurationCapabilities to iSCSIConfigurationService)	Conditional	Conditional requirement: Active configuration is supported.
8.7.8 CIM_ElementSettingData (iSCSIConnectionSettings to TCPProtocolEndpoint)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.9 CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.10 CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.11 CIM_ElementSettingData (iSCSIConnectionSettings to ComputerSystem)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.12 CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.13 CIM_ElementStatisticalData (iSCSILoginStatistics to SCSIProtocolController)	Optional	
8.7.14 CIM_ElementStatisticalData (iSCSIConnectionFailures to SCSIProtocolController)	Optional	
8.7.15 CIM_ElementStatisticalData (iSCSIConnectionStatistics to iSCSIConnection)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.16 CIM_EndpointOfNetworkPipe (iSCSIConnection to TCPProtocolEndpoint)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.17 CIM_EndpointOfNetworkPipe (iSCSIConnection to iSCSIProtocolEndpoint)	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.18 CIM_EthernetPort	Optional	

Table 30 - CIM Elements for iSCSI Target Ports

Element Name	Requirement	Description
8.7.19 CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint)	Mandatory	
8.7.20 CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint)	Mandatory	
8.7.21 CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint)	Mandatory	
8.7.22 CIM_HostedCollection	Conditional	This is required if CIM_SystemSpecificCollection is used.
8.7.23 CIM_HostedService	Optional	
8.7.24 CIM_IPProtocolEndpoint	Mandatory	
8.7.25 CIM_iSCSICapabilities	Mandatory	
8.7.26 CIM_iSCSIConfigurationCapabilities	Conditional	Conditional requirement: Active configuration is supported (iSCSIConfigurationService is supported).
8.7.27 CIM_iSCSIConfigurationService	Optional	Active configuration is supported.
8.7.28 CIM_iSCSIConnection	Optional	
8.7.29 CIM_iSCSIConnectionSettings	Optional	
8.7.30 CIM_iSCSILoginStatistics	Optional	
8.7.31 CIM_iSCSIProtocolEndpoint	Mandatory	
8.7.32 CIM_iSCSISession	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.33 CIM_iSCSISessionFailures	Optional	
8.7.34 CIM_iSCSISessionSettings	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').
8.7.35 CIM_iSCSISessionStatistics	Optional	
8.7.36 CIM_MemberOfCollection	Optional	
8.7.37 CIM_NetworkPipeComposition	Optional	
8.7.38 CIM_SAPAvailableForElement	Mandatory	
8.7.39 CIM_SCSIProtocolController	Mandatory	
8.7.40 CIM_SystemDevice (ComputerSystem to EthernetPort)	Mandatory	This association links all EthernetPorts to the scoping system.
8.7.41 CIM_SystemDevice (ComputerSystem to SCSIProtocolController)	Mandatory	This association links SCSIProtocolControllers to the scoping system.
8.7.42 CIM_SystemSpecificCollection	Optional	
8.7.43 CIM_TCPProtocolEndpoint	Mandatory	
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_EthernetPort	Optional	Create EthernetPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_EthernetPort AND SourceInstance.CIM_EthernetPort::OperationalStatus <> PreviousInstance.CIM_EthernetPort::OperationalStatus	Optional	CQL -Modify EthernetPort.

Table 30 - CIM Elements for iSCSI Target Ports

Element Name	Requirement	Description
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_EthernetPort	Optional	Delete EthernetPort.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_iSCSIProtocolEndpoint	Mandatory	Create iSCSIProtocolEndpoint.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_iSCSIProtocolEndpoint	Mandatory	Delete SCSIProtocolEndpoint.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_SCSIProtocolController	Mandatory	Create SCSIProtocolController.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_SCSIProtocolController	Mandatory	Delete iSCSIProtocolController.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_iSCSISession	Optional	Create iSCSISession.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_iSCSISession AND SourceInstance.CIM_iSCSISession::CurrentConnections <> PreviousInstance.CIM_iSCSISession::CurrentConnections	Optional	CQL -Modify iSCSISession.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_iSCSISession	Optional	Delete iSCSISession.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_iSCSIConnection	Optional	Create iSCSIConnection.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_iSCSIConnection	Optional	Delete iSCSIConnection.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_iSCSISessionSettings	Conditional	Conditional requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session'). Modify iSCSISessionSettings.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_iSCSIConnectionSettings	Optional	Modify iSCSIConnectionSettings.

8.7.1 CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 31 describes class CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint).

Table 31 - SMI Referenced Properties/Methods for CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to IPProtocolEndpoint
Dependent		Mandatory	Reference to TCPProtocolEndpoint

8.7.2 CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 32 describes class CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint).

Table 32 - SMI Referenced Properties/Methods for CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to TCPProtocolEndpoint
Dependent		Mandatory	Reference to iSCSIProtocolEndpoint

8.7.3 CIM_ConcreteDependency

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Conditional. This is required if CIM_SystemSpecificCollection is used.

Table 33 describes class CIM_ConcreteDependency.

Table 33 - SMI Referenced Properties/Methods for CIM_ConcreteDependency

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to SystemSpecificCollection

8.7.4 CIM_DeviceSAPImplementation (EthernetPort to IPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 34 describes class CIM_DeviceSAPImplementation (EthernetPort to IPProtocolEndpoint).

Table 34 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (EthernetPort to IPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to EthernetPort
Dependent		Mandatory	Reference to IPProtocolEndpoint

8.7.5 CIM_DeviceSAPImplementation (EthernetPort to iSCSIProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 35 describes class CIM_DeviceSAPImplementation (EthernetPort to iSCSIProtocolEndpoint).

Table 35 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (EthernetPort to iSCSIProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to EthernetPort
Dependent		Mandatory	Reference to iSCSIProtocolEndpoint

8.7.6 CIM_ElementCapabilities (CIM_iSCSICapabilities to System)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 36 describes class CIM_ElementCapabilities (CIM_iSCSICapabilities to ComputerSystem).

Table 36 - SMI Referenced Properties/Methods for CIM_ElementCapabilities (CIM_iSCSICapabilities to ComputerSystem)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to ComputerSystem
Capabilities		Mandatory	Reference to iSCSICapabilities

8.7.7 CIM_ElementCapabilities (iSCSIConfigurationCapabilities to iSCSIConfigurationService)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Active configuration is supported.

Table 37 describes class CIM_ElementCapabilities (iSCSIConfigurationCapabilities to iSCSIConfigurationService).

Table 37 - SMI Referenced Properties/Methods for CIM_ElementCapabilities (iSCSIConfigurationCapabilities to iSCSIConfigurationService)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to iSCSIConfigurationService
Capabilities		Mandatory	Reference to iSCSIConfigurationCapabilities

8.7.8 CIM_ElementSettingData (iSCSIConnectionSettings to TCPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 38 describes class CIM_ElementSettingData (iSCSIConnectionSettings to TCPProtocolEndpoint).

Table 38 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSIConnectionSettings to TCPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to TCPProtocolEndpoint
SettingData		Mandatory	Reference to iSCSIConnectionSettings

8.7.9 CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 39 describes class CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint).

Table 39 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSIConnectionSettings to iSCSIProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to iSCSIProtocolEndpoint
SettingData		Mandatory	Reference to iSCSIConnectionSettings

8.7.10 CIM_ElementSettingData (iSCSISessionSettings to SCSIProtocolController)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 40 describes class CIM_ElementSettingData (iSCSISessionSettings to SCSIProtocolController).

Table 40 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSISessionSettings to SCSIProtocolController)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to SCSIProtocolController
SettingData		Mandatory	Reference to iSCSISessionSettings

8.7.11 CIM_ElementSettingData (iSCSI_SessionSettings to ComputerSystem)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 41 describes class CIM_ElementSettingData (iSCSI_SessionSettings to ComputerSystem).

Table 41 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSI_SessionSettings to ComputerSystem)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to ComputerSystem
SettingData		Mandatory	Reference to iSCSI_SessionSettings

8.7.12 CIM_ElementSettingData (iSCSI_SessionSettings to iSCSI_ProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 42 describes class CIM_ElementSettingData (iSCSI_SessionSettings to iSCSI_ProtocolEndpoint).

Table 42 - SMI Referenced Properties/Methods for CIM_ElementSettingData (iSCSI_SessionSettings to iSCSI_ProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to iSCSI_ProtocolEndpoint
SettingData		Mandatory	Reference to iSCSI_SessionSettings

8.7.13 CIM_ElementStatisticalData (iSCSI_LoginStatistics to SCSI_ProtocolController)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 43 describes class CIM_ElementStatisticalData (iSCSI_LoginStatistics to SCSI_ProtocolController).

Table 43 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (iSCSI_LoginStatistics to SCSI_ProtocolController)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to iSCSI_ProtocolEndpoint
Stats		Mandatory	Reference to iSCSI_LoginStatistics

8.7.14 CIM_ElementStatisticalData (iSCSI_SessionFailures to SCSIProtocolController)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 44 describes class CIM_ElementStatisticalData (iSCSI_SessionFailures to SCSIProtocolController).

Table 44 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (iSCSI_SessionFailures to SCSIProtocolController)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to iSCSIProtocolEndpoint
Stats		Mandatory	Reference to iSCSI_SessionFailures

8.7.15 CIM_ElementStatisticalData (iSCSI_SessionStatistics to iSCSI_Session)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 45 describes class CIM_ElementStatisticalData (iSCSI_SessionStatistics to iSCSI_Session).

Table 45 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (iSCSI_SessionStatistics to iSCSI_Session)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to iSCSI_Session
Stats		Mandatory	Reference to iSCSI_SessionStatistics

8.7.16 CIM_EndpointOfNetworkPipe (iSCSI_Connection to TCPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 46 describes class CIM_EndpointOfNetworkPipe (iSCSI_Connection to TCPProtocolEndpoint).

Table 46 - SMI Referenced Properties/Methods for CIM_EndpointOfNetworkPipe (iSCSI_Connection to TCP-ProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to TCPProtocolEndpoint
Dependent		Mandatory	Reference to iSCSI_Connection

8.7.17 CIM_EndpointOfNetworkPipe (iSCSISession to iSCSIProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 47 describes class CIM_EndpointOfNetworkPipe (iSCSISession to iSCSIProtocolEndpoint).

Table 47 - SMI Referenced Properties/Methods for CIM_EndpointOfNetworkPipe (iSCSISession to iSCSIProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to iSCSIProtocolEndpoint
Dependent		Mandatory	Reference to iSCSISession

8.7.18 CIM_EthernetPort

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 48 describes class CIM_EthernetPort.

Table 48 - SMI Referenced Properties/Methods for CIM_EthernetPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
PermanentAddress	CD	Mandatory	

8.7.19 CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 49 describes class CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint).

Table 49 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to IPProtocolEndpoint

8.7.20 CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 50 describes class CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint).

Table 50 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to TCPProtocolEndpoint

8.7.21 CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 51 describes class CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint).

Table 51 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to iSCSIProtocolEndpoint

8.7.22 CIM_HostedCollection

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Conditional. This is required if CIM_SystemSpecificCollection is used.

Table 52 describes class CIM_HostedCollection.

Table 52 - SMI Referenced Properties/Methods for CIM_HostedCollection

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to iSCSIPortalGroup (SystemSpecificCollection)

8.7.23 CIM_HostedService

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 53 describes class CIM_HostedService.

Table 53 - SMI Referenced Properties/Methods for CIM_HostedService

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to iSCSIConfigurationService

8.7.24 CIM_IPProtocolEndpoint

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 54 describes class CIM_IPProtocolEndpoint.

Table 54 - SMI Referenced Properties/Methods for CIM_IPProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
IPv4Address	CD	Optional	
IPv6Address	CD	Optional	
ProtocolIFType		Mandatory	

8.7.25 CIM_iSCSICapabilities

Created By: Static

Modified By: Static

Deleted By: Static
Requirement: Mandatory

Table 55 describes class CIM_iSCSICapabilities.

Table 55 - SMI Referenced Properties/Methods for CIM_iSCSICapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
MinimumSpecificationVersionSupported		Mandatory	
MaximumSpecificationVersionSupported		Mandatory	
AuthenticationMethodsSupported		Mandatory	
SupportedFeatures		Mandatory	
OtherSupportedFeatures		Conditional	Required when the corresponding SupportedFeatures array element is set to the value "1" (Other). Each element that corresponds to a SupportedFeatures array element with a value other than "1" (Other) shall have a value of NULL

8.7.26 CIM_iSCSIConfigurationCapabilities

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Active configuration is supported (iSCSIConfigurationService is implemented).

Table 56 describes class CIM_iSCSIConfigurationCapabilities.

Table 56 - SMI Referenced Properties/Methods for CIM_iSCSIConfigurationCapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
iSCSINodeCreationSupported		Mandatory	
iSCSIProtocolEndpointCreationSupported		Mandatory	
IdentifierSelectionSupported		Mandatory	

8.7.27 CIM_iSCSIConfigurationService

Created By: Static
Modified By: Static
Deleted By: Static

Requirement: Optional

Table 57 describes class CIM_iSCSIConfigurationService.

Table 57 - SMI Referenced Properties/Methods for CIM_iSCSIConfigurationService

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	

8.7.28 CIM_iSCSIConnection

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 58 describes class CIM_iSCSIConnection.

Table 58 - SMI Referenced Properties/Methods for CIM_iSCSIConnection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ConnectionID		Mandatory	
MaxReceiveDataSegment Length		Mandatory	
MaxTransmitDataSegment Length		Mandatory	
HeaderDigestMethod		Mandatory	
OtherHeaderDigestMethod		Conditional	Required when HeaderDigestMethod is Other
DataDigestMethod		Mandatory	
OtherDataDigestMethod		Conditional	Required when DataDigestMethod is Other
ReceivingMarkers		Mandatory	
SendingMarkers		Mandatory	
ActiveiSCSIVersion		Mandatory	
AuthenticationMethodUsed		Mandatory	
MutualAuthentication		Mandatory	

8.7.29 CIM_iSCSIConnectionSettings

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 59 describes class CIM_iSCSIConnectionSettings.

Table 59 - SMI Referenced Properties/Methods for CIM_iSCSIConnectionSettings

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
MaxReceiveDataSegment Length		Mandatory	
PrimaryHeaderDigestMethod		Mandatory	
OtherPrimaryHeaderDigestMethod		Conditional	Required when PrimaryHeaderDigestMethod is Other
PrimaryDataDigestMethod		Mandatory	
OtherPrimaryDataDigestMethod		Conditional	Required when PrimaryDataDigestMethod is Other
SecondaryHeaderDigestMethod		Mandatory	
OtherSecondaryHeaderDigestMethod		Conditional	Required when SecondaryHeaderDigestMethod is Other
SecondaryDataDigestMethod		Mandatory	
OtherSecondaryDataDigestMethod		Conditional	Required when SecondaryDataDigestMethod is Other
RequestingMarkersOnReceive		Mandatory	
PrimaryAuthenticationMethod		Mandatory	
SecondaryAuthenticationMethod		Mandatory	

8.7.30 CIM_iSCSILoginStatistics

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 60 describes class CIM_iSCSILoginStatistics.

Table 60 - SMI Referenced Properties/Methods for CIM_iSCSILoginStatistics

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	

Table 60 - SMI Referenced Properties/Methods for CIM_iSCSILoginStatistics

Properties	Flags	Requirement	Description & Notes
LastSessionFailureType		Optional	
OtherLastLoginFailureType		Conditional	Required when LastLoginFailureType is Other

8.7.31 CIM_iSCSIProtocolEndpoint

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 61 describes class CIM_iSCSIProtocolEndpoint.

Table 61 - SMI Referenced Properties/Methods for CIM_iSCSIProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	CD	Mandatory	MUST contain the iSCSI port name as described in the iSCSI RFC (currently http://www.ietf.org/rfc/rfc3720.txt) - < iSCSI node name> + ',t,' + TPGT for targets
ConnectionType		Mandatory	iSCSI.
Identifier		Mandatory	Must contain the TPGT.
ProtocolType		Mandatory	Other.
OtherTypeDescription		Mandatory	Must be "iSCSI"
Role		Mandatory	Shall be 3 (Target) or 4 (Both Initiator and Target).

8.7.32 CIM_iSCSISession

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 62 describes class CIM_iSCSISession.

Table 62 - SMI Referenced Properties/Methods for CIM_iSCSISession

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
Directionality		Mandatory	
SessionType		Mandatory	

Table 62 - SMI Referenced Properties/Methods for CIM_iSCSISession

Properties	Flags	Requirement	Description & Notes
TSIH		Mandatory	
EndPointName		Mandatory	
CurrentConnections		Mandatory	
InitialR2T		Mandatory	
ImmediateData		Mandatory	
MaxOutstandingR2T		Mandatory	
MaxUnsolicitedFirstDataBurstLength		Mandatory	
MaxDataBurstLength		Mandatory	
DataSequenceInOrder		Mandatory	
DataPDUInOrder		Mandatory	
ErrorRecoveryLevel		Mandatory	
MaxConnectionsPerSession		Mandatory	
DefaultTimeToWait		Mandatory	
DefaultTimeToRetain		Mandatory	

8.7.33 CIM_iSCSISessionFailures

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 63 describes class CIM_iSCSISessionFailures.

Table 63 - SMI Referenced Properties/Methods for CIM_iSCSISessionFailures

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
LastSessionFailureType		Optional	
OtherLastSessionFailureType		Conditional	Required when LastSessionFailureType is Other

8.7.34 CIM_iSCSISessionSettings

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: This is required if CIM_iSCSICapabilities.SupportedFeatures = '3' ('iSCSI Session').

Table 64 describes class CIM_iSCSISessionSettings.

Table 64 - SMI Referenced Properties/Methods for CIM_iSCSISessionSettings

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
MaxConnectionsPerSession		Mandatory	
InitialR2TPreference		Mandatory	
ImmediateDataPreference		Mandatory	
MaxOutstandingR2T		Mandatory	
MaxUnsolicitedFirstDataBurstLength		Mandatory	
MaxDataBurstLength		Mandatory	
DataSequenceInOrderPreference		Mandatory	
DataPDUInOrderPreference		Mandatory	
DefaultTimeToWaitPreference		Mandatory	
DefaultTimeToRetainPreference		Mandatory	
ErrorRecoveryLevelPreference		Mandatory	

8.7.35 CIM_iSCSISessionStatistics

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 65 describes class CIM_iSCSISessionStatistics.

Table 65 - SMI Referenced Properties/Methods for CIM_iSCSISessionStatistics

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	

8.7.36 CIM_MemberOfCollection

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 66 describes class CIM_MemberOfCollection.

Table 66 - SMI Referenced Properties/Methods for CIM_MemberOfCollection

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	Reference to iSCSIPortalGroup (SystemSpecificCollection
Member		Mandatory	Reference to TCPProtocolEndpoint

8.7.37 CIM_NetworkPipeComposition

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 67 describes class CIM_NetworkPipeComposition.

Table 67 - SMI Referenced Properties/Methods for CIM_NetworkPipeComposition

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to iSCSIConnection
PartComponent		Mandatory	Reference to iSCSISession

8.7.38 CIM_SAPAvailableForElement

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 68 describes class CIM_SAPAvailableForElement.

Table 68 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to EthernetPort
AvailableSAP		Mandatory	Reference to IPProtocolEndpoint

8.7.39 CIM_SCSIProtocolController

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 69 describes class CIM_SCSIProtocolController.

Table 69 - SMI Referenced Properties/Methods for CIM_SCSIProtocolController

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
ElementName		Mandatory	iSCSI Alias.
Name	CD	Mandatory	
NameFormat		Mandatory	Must be '3' iSCSI Name

8.7.40 CIM_SystemDevice (ComputerSystem to EthernetPort)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 70 describes class CIM_SystemDevice (ComputerSystem to EthernetPort).

Table 70 - SMI Referenced Properties/Methods for CIM_SystemDevice (ComputerSystem to EthernetPort)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem
PartComponent		Mandatory	Reference to EthernetPort

8.7.41 CIM_SystemDevice (ComputerSystem to SCSIProtocolController)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 71 describes class CIM_SystemDevice (ComputerSystem to SCSIProtocolController).

Table 71 - SMI Referenced Properties/Methods for CIM_SystemDevice (ComputerSystem to SCSIProtocolController)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem
PartComponent		Mandatory	Reference to SCSIProtocolController

8.7.42 CIM_SystemSpecificCollection

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 72 describes class CIM_SystemSpecificCollection (iSCSIPortalGroup).

Table 72 - SMI Referenced Properties/Methods for CIM_SystemSpecificCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	

8.7.43 CIM_TCPProtocolEndpoint

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 73 describes class CIM_TCPProtocolEndpoint.

Table 73 - SMI Referenced Properties/Methods for CIM_TCPProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
PortNumber		Mandatory	
ProtocolIFType		Mandatory	

STABLE

iSCSI Target Ports Profile

STABLE

9 Serial Attached SCSI (SAS) Target Port Profile

9.1 Synopsis

Profile Name: SAS Target Ports (Component Profile)

Version: 1.7.0

Organization: SNIA

Central Class: CIM_SASPort

Scoping Class: a CIM_System in a separate autonomous profile

Related Profiles: Table 74 describes the related profiles for SAS Target Ports.

Table 74 - Related Profiles for SAS Target Ports

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

9.2 Description

Figure 12 illustrates the Serial Attached SCSI (SAS) Target Port. Serial Attached SCSI is a lower cost network interface for SCSI communication.

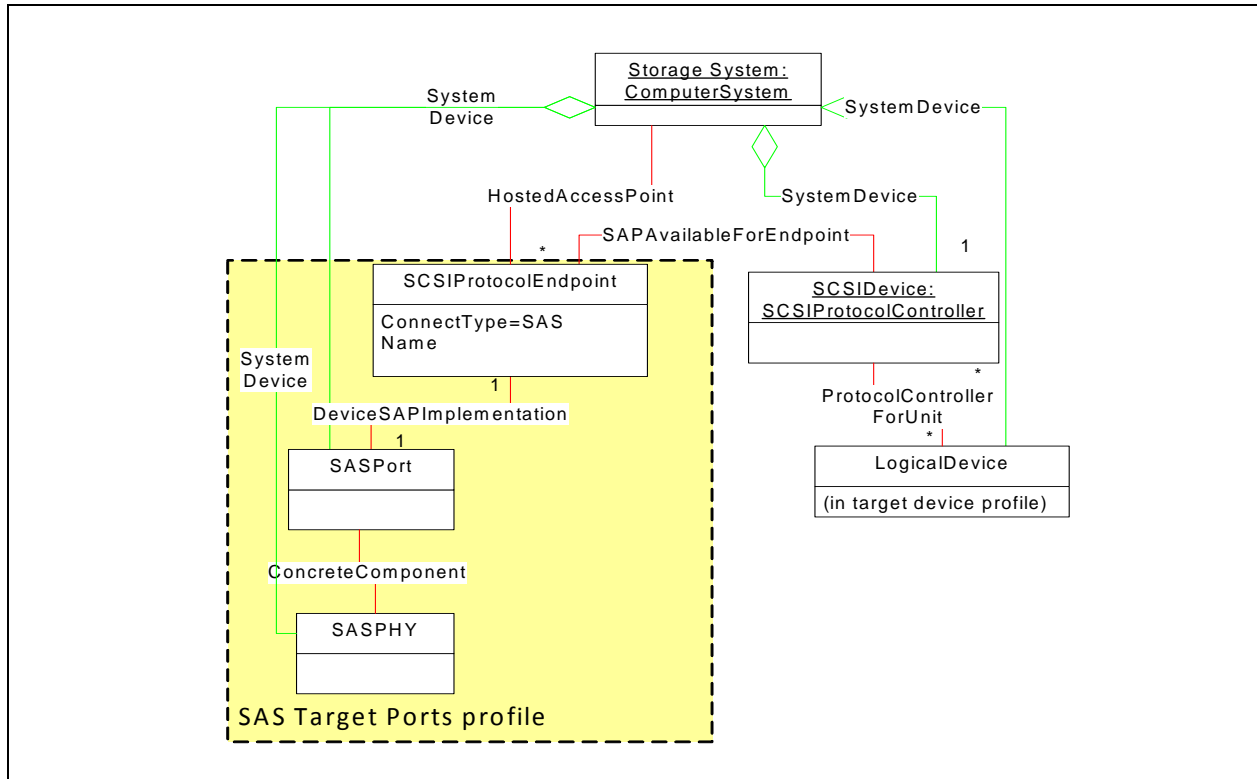


Figure 12 - Serial Attached SCSI (SAS) Target Port Instance Diagram

SCSIProtocolEndpoint.ConnectionType shall be set to "SAS". SASPort represents the port and is connected to SCSIProtocolEndpoint by DeviceSAPImplementation. The SASPort contains information about the speed for the bus.

The SASPHY class represents a SAS PHY. A SAS Port may have multiple associated PHYs; generally, all the PHYs are connected to the same initiator or expander and provide additional bandwidth.

9.2.1 Health and Fault Management

Table 75 describes SASPort OperationalStatus.

Table 75 - SASPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

9.3 Methods

9.3.1 Extrinsic Methods of this Profile

Not defined in this standard

9.3.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

9.4 Client Considerations and Recipes

No recipes are defined in this version of the standard.

9.5 CIM Elements

Table 76 describes the CIM elements for SAS Target Ports.

Table 76 - CIM Elements for SAS Target Ports

Element Name	Requirement	Description
9.5.1 CIM_ConcreteComponent	Mandatory	Associates SASPort and SASPHY.
9.5.2 CIM_DeviceSAPImplementation	Mandatory	Associates SASPort and SCSIProtocolEndpoint.
9.5.3 CIM_HostedAccessPoint	Mandatory	Associates ComputerSystem to ProtocolEndpoint.
9.5.4 CIM_SASPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
9.5.5 CIM_SCSIProtocolEndpoint	Mandatory	Represents management characteristics related to the SCSI command set.
9.5.6 CIM_SystemDevice (Port)	Mandatory	Associates ComputerSystem to LogicalPort.
9.5.7 CIM_SystemDevice (SAS PHY)	Mandatory	Associates ComputerSystem to SASPHY.
9.5.8 CIM_SASPHY	Mandatory	Several SASPHYs may together be aggregated into a SAS Logical Port.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_SASPort	Mandatory	CQL -Create SASPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_SASPort AND SourceInstance.CIM_SASPort::OperationalStatus <> PreviousInstance.SAS_Port::OperationalStatus	Mandatory	CQL -Modify SASPort.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_SASPort	Mandatory	CQL -Delete SASPort.

9.5.1 CIM_ConcreteComponent

Associates SASPort and SASPHY.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 77 describes class CIM_ConcreteComponent.

Table 77 - SMI Referenced Properties/Methods for CIM_ConcreteComponent

Properties	Flags	Requirement	Description & Notes
PartComponent		Mandatory	Reference to SASPHY.
GroupComponent		Mandatory	Reference to SASPort.

9.5.2 CIM_DeviceSAPImplementation

Associates SASPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 78 describes class CIM_DeviceSAPImplementation.

Table 78 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.
Antecedent		Mandatory	Reference to SASPort.

9.5.3 CIM_HostedAccessPoint

Associates ComputerSystem to ProtocolEndpoint. Limit to targets. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Target Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 79 describes class CIM_HostedAccessPoint.

Table 79 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in the referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.

9.5.4 CIM_SASPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Target Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 80 describes class CIM_SASPort.

Table 80 - SMI Referenced Properties/Methods for CIM_SASPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	

Table 80 - SMI Referenced Properties/Methods for CIM_SASPort

Properties	Flags	Requirement	Description & Notes
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 2 (Front-end Only).
PortType		Mandatory	Shall be 94 (SAS).
PermanentAddress		Mandatory	SAS Address. Shall be 16 un-separated upper case hex digits.

9.5.5 CIM_SCSIProtocolEndpoint

Represents management characteristics related to the SCSI command set. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 81 describes class CIM_SCSIProtocolEndpoint.

Table 81 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 8 (SAS).
Role		Mandatory	Shall be 3 (Target).

9.5.6 CIM_SystemDevice (Port)

Associates ComputerSystem to LogicalPort. The class definition specializes the CIM_SystemDevice definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 82 describes class CIM_SystemDevice (Port).

Table 82 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem in the referencing profile.
PartComponent		Mandatory	Reference to SASPort.

9.5.7 CIM_SystemDevice (SAS PHY)

Associates ComputerSystem to SASPHY.

Requirement: Mandatory

Table 83 describes class CIM_SystemDevice (SAS PHY).

Table 83 - SMI Referenced Properties/Methods for CIM_SystemDevice (SAS PHY)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to SASPHY.

9.5.8 CIM_SASPHY

Several SASPHYs may together be aggregated into a SAS Logical Port.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 84 describes class CIM_SASPHY.

Table 84 - SMI Referenced Properties/Methods for CIM_SASPHY

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
HardwareMinimumPhysicalLinkRate		Mandatory	
HardwareMaximumPhysicalLinkRate		Mandatory	
ProgrammedMinimumPhysicalLinkRate		Mandatory	
ProgrammedMaximumPhysicalLinkRate		Mandatory	
NegotiatedPhysicalLinkRate		Mandatory	

STABLE

DEPRECATED

10 Serial ATA (SATA) Target Ports Profile

The functionality of the Experimental SATA Target Ports Profile has been removed.

The Serial ATA Target Ports Profile is defined in Clause 11 of SMI-S 1.6.1

DEPRECATED

EXPERIMENTAL

11 SB Target Ports Profile

11.1 Synopsis

Profile Name: SB Target Ports (Component Profile)

Version: 1.7.0

Organization: SNIA

Central Class: FCPort

Scoping Class: ComputerSystem

Related Profiles: Table 85 describes the related profiles for SB Target Ports.

Table 85 - Related Profiles for SB Target Ports

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

11.2 Description

The SB Target Port Profile models the SB (Single Byte) Fibre Channel specific aspects of a target storage system. The Single Byte protocols are FC4 protocols that support mainframe IO (as opposed to SCSI, which supports IO from non-mainframe systems such as Unix or Windows systems).

The SB Target Port Profile provides a way for storage profiles to model target ports that are dedicated to serving SB hosts attachment. With this support a client will be able to distinguish FC ports that are provided for SCSI access from FC Ports that are provided for mainframe attachment. This is an important distinction for management, since fabric connectivity collections for SB would typically be separate for fabric connectivity collections for SCSI. Similarly, management functions for masking and mapping are somewhat different for SB than SCSI. So, it is important for management applications to be aware of the distinctions.

The SB Target Port Profile specializes the Generic Target Port Profile.

For SB enabled Fibre Channel ports, the concrete subclass of LogicalPort is FCPort. FCPort is always associated 1-1 with a SBProtocolEndpoint instance.

11.3 Implementation

Figure 13: "SB Target Port Instance Diagram" illustrates the SB Target Port Profile.

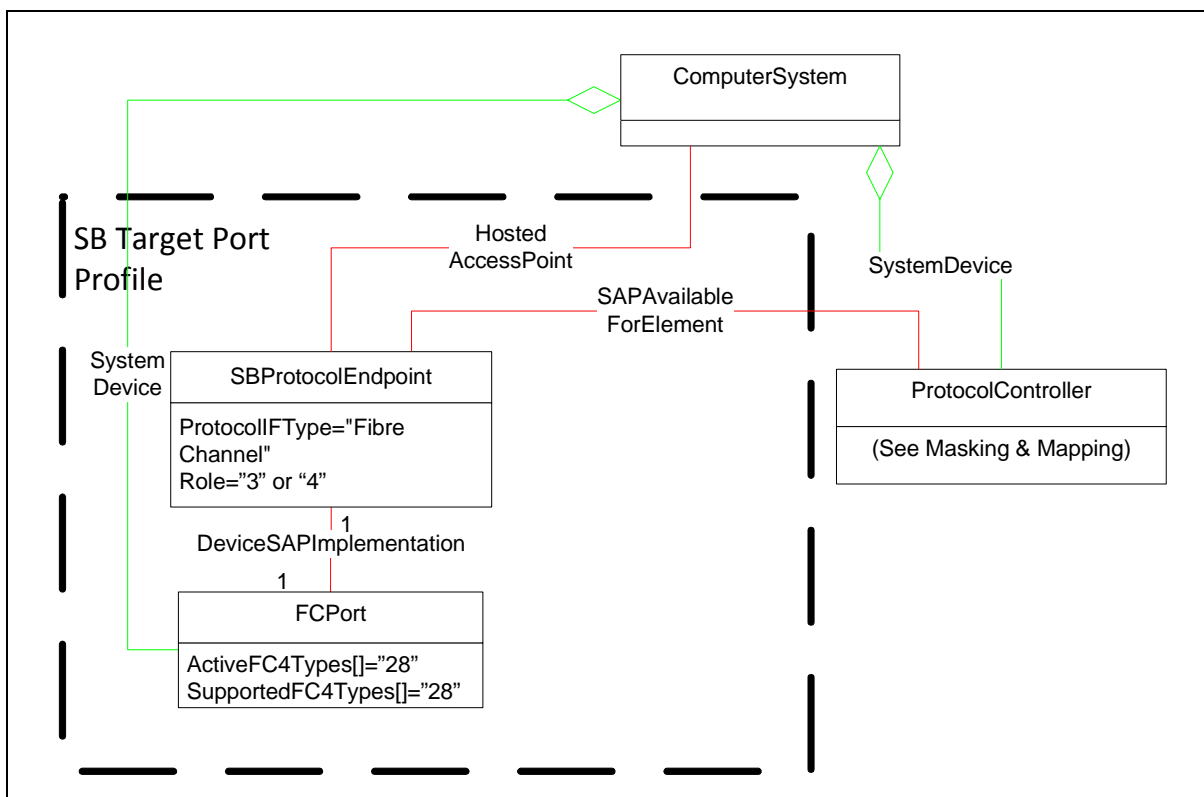


Figure 13 - SB Target Port Instance Diagram

SB Ports are Fibre Channel Ports with the `SupportedFC4Types[]` and `ActiveFC4Types[]` arrays holding the value "28" (for "FC-SB-2 Control Unit"). The `SupportedFC4Types[]` property shall contain the value "28". The `ActiveFC4Types[]` property shall contain the value "28" for FCPorts that are actively supporting SB protocols.

The FCPort shall also support an SBProtocolEndpoint with a role property of either "3" ("Target") or "4" ("Both initiator and target").

For the SB Target Port Profile, the FCPort is the central class of the Profile.

11.4 Health and Fault Management Consideration

Table 86 defines the SMI-S defined meanings of the `OperationalStatus` property for FCPorts used in the SB Target Port Profile.

Table 86 - FCPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in self test
Unknown	

11.5 Cascading Considerations

Not defined in this standard

11.6 Methods of the Profile

11.6.1 Extrinsic Methods of the Profile

Not defined in this standard

11.6.2 Intrinsic Methods of the Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

11.7 Client Considerations and Recipes

No recipes are defined in this version of the standard.

11.8 CIM Elements

Table 87 describes the CIM elements for SB Target Ports.

Table 87 - CIM Elements for SB Target Ports

Element Name	Requirement	Description
11.8.1 CIM_DeviceSAPImplementation	Mandatory	Associates front-end FCPort and target SBProtocolEndpoint.
11.8.2 CIM_FCPort	Mandatory	The FCPort provides transport for SB protocols.
11.8.3 CIM_HostedAccessPoint	Mandatory	Associates ComputerSystem to SBProtocolEndpoint.
11.8.4 CIM_SystemDevice (Port)	Mandatory	Associates controller ComputerSystem to FCPort.
11.8.5 CIM_SBProtocolEndpoint	Mandatory	This defines the protocol being used on the FC Port as SB (Single Byte).
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_FCPort	Mandatory	Create FCPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::OperationalStatus <> PreviousInstance.CIM_FCPort::OperationalStatus	Mandatory	CQL -Change to FCPort OperationalStatus.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FCPort	Mandatory	Delete FCPort.

11.8.1 CIM_DeviceSAPImplementation

Associates front-end FCPort and target SBProtocolEndpoint. Limit to target ProtocolEndpoints and front-end ports. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 88 describes class CIM_DeviceSAPImplementation.

Table 88 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to FCPort
Dependent		Mandatory	Reference to SBProtocolEndpoint

11.8.2 CIM_FCPort

The CIM_FCPort class for SB Target Ports is the same as the FC Target Port class, except that the SupportedFC4Types and ActiveFC4Types properties are mandatory, and the SupportedFC4Types shall contain "28" and the ActiveFC4Types should contain "28" to indicate support for FC-SB-2 Control Unit functions. The class definition specializes the CIM_LogicalPort definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 89 describes class CIM_FCPort.

Table 89 - SMI Referenced Properties/Methods for CIM_FCPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 2 (Front-end Only).
OtherPortType		Conditional	Required when PortType is Other
PermanentAddress	CD	Mandatory	Port WWN. Shall be 16 unseparated uppercase hex digits.
SupportedFC4Types		Mandatory	For SB Target Ports this array shall contain 28 (FC-SB-2 Control Unit).
ActiveFC4Types		Mandatory	For SB Target Ports this array should contain 28 (FC-SB-2 Control Unit).

11.8.3 CIM_HostedAccessPoint

Associates ComputerSystem to SBProtocolEndpoint. Limit to targets (Role = 3). The class definition specializes the CIM_HostedAccessPoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 90 describes class CIM_HostedAccessPoint.

Table 90 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in the referencing profile.
Dependent		Mandatory	Reference to SBProtocolEndpoint.

11.8.4 CIM_SystemDevice (Port)

Associates controller ComputerSystem to FCPort. The class definition specializes the CIM_SystemDevice definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 91 describes class CIM_SystemDevice (Port).

Table 91 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem in the referencing profile.
PartComponent		Mandatory	Reference to FCPort.

11.8.5 CIM_SBProtocolEndpoint

The CIM_SBProtocolEndpoint specializes the Generic Target Port CIM_ProtocolEndpoint. The main difference is that the OtherTypeDescription is set to "SB". The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 92 describes class CIM_SBProtocolEndpoint.

Table 92 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SB'.
Role		Mandatory	Shall be 3 (Target).
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).

EXPERIMENTAL

STABLE

12 Direct Attach (DA) Ports Profile

12.1 Synopsis

Profile Name: Direct Access Ports (Component Profile)

Version: 1.4.0

Organization: SNIA

Central Class: DAPort

Scoping Class: ComputerSystem in a separate autonomous profile

Specializes: SNIA Generic Target Ports version 1.4.0

Related Profiles: Not defined in this standard.

12.2 Description

The DAPort (Direct Attach) port models storage systems that attach directly to buses in a host system (e.g., PCI, PCI-E, and chip interfaces on a motherboard). The DAPort can be viewed as both the initiator and Target ports.

This port can not be used with the LUN Mapping and Masking Profile. All volumes served by this port are fully accessible by the host system.

Figure 14 illustrates the Direct Attach (DA) Ports Profile. Volumes served by this port shall be discovered and presented by the Host Discovered Resources Profile.

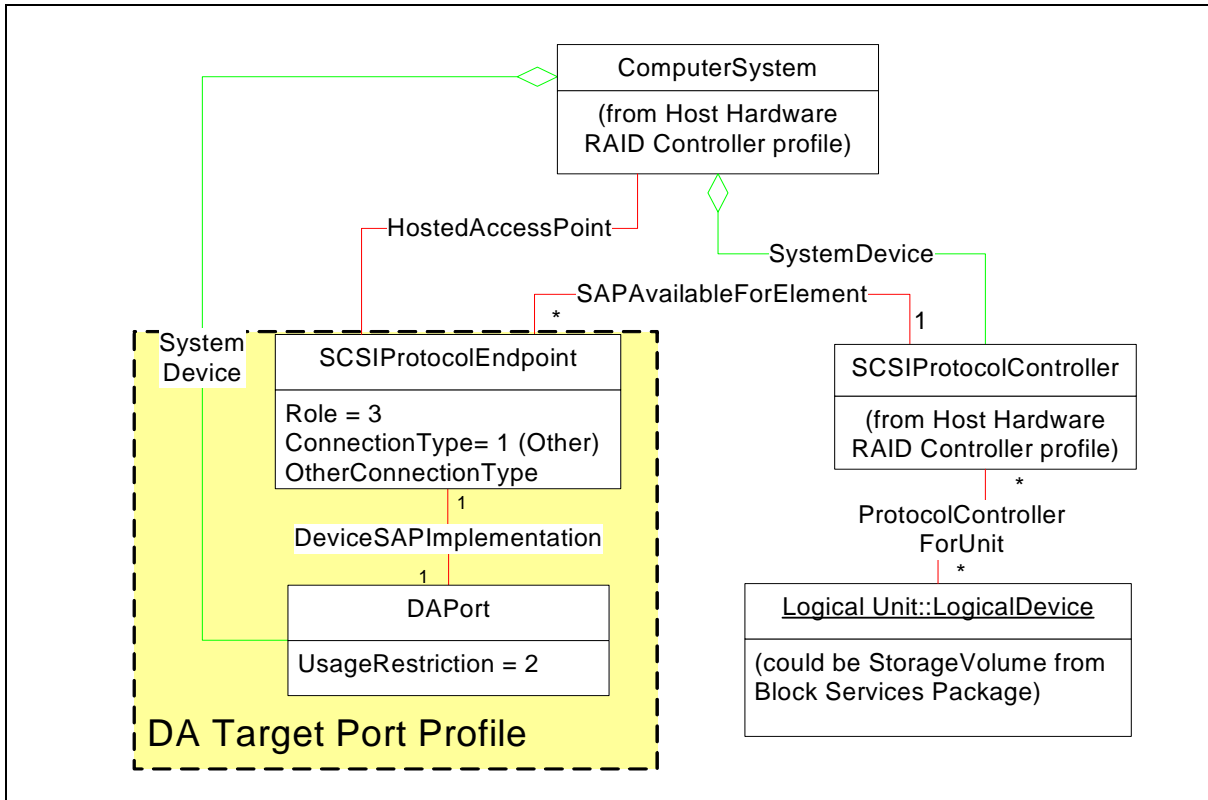


Figure 14 - DA Port Instance Diagram

The DAPort class is connected to the ProtocolEndpoint using the DeviceSAPImplementation association and to the controller ComputerSystem (in the Host Hardware RAID Controller Profile) using SystemDevice. SCSIProtocolController is associated to the controller ComputerSystem from the same ComputerSystem using HostedAccessPoint.

12.3 Health and Fault Management

Not defined in this standard.

12.4 Extrinsic Methods

Not defined in this standard.

12.5 Use Cases

Not defined in this version of the standard.

12.6 CIM Elements

Table 93 describes the CIM elements for DA Target Ports.

Table 93 - CIM Elements for DA Target Ports

Element Name	Requirement	Description
12.6.1 CIM_DAPort	Mandatory	Models the "port" emulated by a host RAID controller.
12.6.2 CIM_DeviceSAPImplementation	Mandatory	Associates DAPort and SCSIProtocolEndpoint.

Table 93 - CIM Elements for DA Target Ports

Element Name	Requirement	Description
12.6.3 CIM_HostedAccessPoint	Mandatory	Associates ComputerSystem to SCSIProtocolEndpoint.
12.6.4 CIM_SCSIProtocolEndpoint	Mandatory	ProtocolEndpoint representing support for SCSI.
12.6.5 CIM_SystemDevice (Port)	Mandatory	Associates controller ComputerSystem to DAPort.

12.6.1 CIM_DAPort

Models the "port" emulated by a host RAID controller. The class definition specializes the CIM_LogicalPort definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 94 describes class CIM_DAPort.

Table 94 - SMI Referenced Properties/Methods for CIM_DAPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 2 (Front-end Only).
PortType		Optional	Not defined in this version of the standard.

12.6.2 CIM_DeviceSAPImplementation

Associates DAPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 95 describes class CIM_DeviceSAPImplementation.

Table 95 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.
Antecedent		Mandatory	Reference to DAPort.

12.6.3 CIM_HostedAccessPoint

Associates ComputerSystem to SCSIProtocolEndpoint. Limit to targets. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 96 describes class CIM_HostedAccessPoint.

Table 96 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to controller ComputerSystem.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.

12.6.4 CIM_SCSIProtocolEndpoint

ProtocolEndpoint representing support for SCSI. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 97 describes class CIM_SCSIProtocolEndpoint.

Table 97 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
Role		Mandatory	Shall be 3 (Target).
ConnectionType		Mandatory	Shall be 1 (Other).
OtherConnectionType		Mandatory	

12.6.5 CIM_SystemDevice (Port)

Associates controller ComputerSystem to DAPort. The class definition specializes the CIM_SystemDevice definition in the Generic Target Ports profile.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 98 describes class CIM_SystemDevice (Port).

Table 98 - SMI Referenced Properties/Methods for CIM_SystemDevice (Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to controller ComputerSystem.
PartComponent		Mandatory	Reference to DAPort.

STABLE

Direct Attach (DA) Ports Profile

IMPLEMENTED

13 Generic Initiator Ports Profile

13.1 Synopsis

Profile name: Generic Initiator Ports

Version: 1.7.0

Organization: SNIA

Central Class: LogicalPort

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Table 99 describes the related profiles for Generic Initiator Ports.

Table 99 - Related Profiles for Generic Initiator Ports

Profile Name	Organization	Version	Requirement	Description
iSCSI Initiator Ports	SNIA	1.2.0		
FC Initiator Ports	SNIA	1.7.0		
SAS Initiator Ports	SNIA	1.7.0		
SB Initiator Ports	SNIA	1.7.0		Experimental.
FCoE Initiator Ports	SNIA	1.6.0		Experimental.

The Generic Initiator Port Profile models the generic management interfaces of initiator ports in host adaptors or storage systems.

This abstract profile specification shall not be directly implemented; implementations shall be based on a profile specification that specializes the requirements of this profile.

13.2 Description

The Generic Initiator Port Profile models the generic behavior of initiator ports in host adaptors. It uses the same primary classes as the Generic Target Port Profile (see 5 Generic Target Ports Profile)

13.3 Implementation

The initiator port is modeled as a ProtocolEndpoint connected to a LogicalPort.

The LogicalDevice instances may represent local storage (embedded in the system containing the initiator ports) or remote storage. When it represents remote storage the Name and NameFormat properties are used as correlatable ids to reference the remote device. When the LogicalDevice represents local disk storage, it may be represented as an instance of StorageVolume (subclass of LogicalDevice) or part of an instance of the Disk Drive Lite Profile. A property on LogicalPort called UsageRestriction is available to indicate whether the controller is capable of providing a “front end” (target), a “back end” (initiator), or both interfaces.

Figure 15: "Generic Initiator Port Model" depicts the generic model

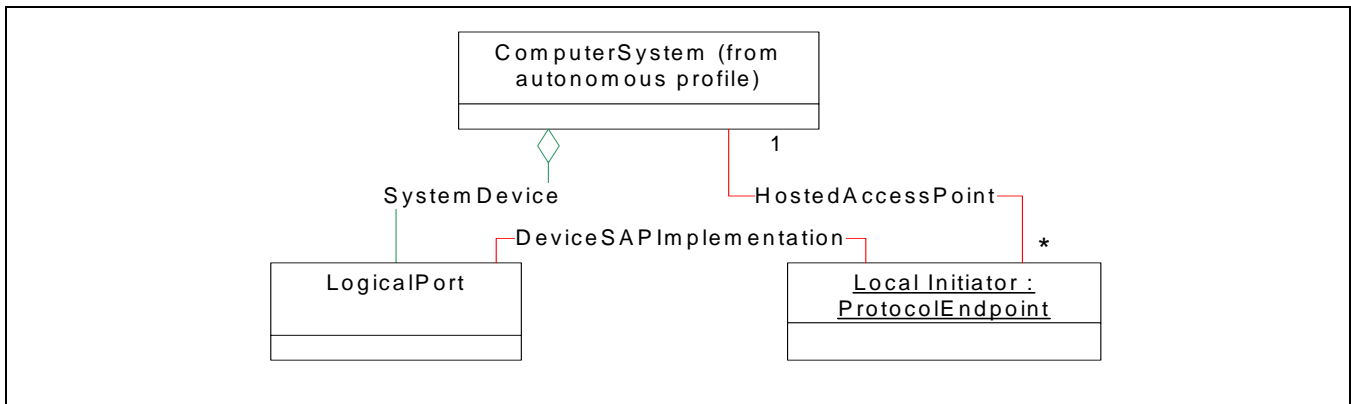


Figure 15 - Generic Initiator Port Model

13.3.1 Remote Device Models

The implementation may optionally include discovered remote elements. There are two optional approaches to modeling remote elements, depending on the capabilities of the underlying host drivers

The first approach is to model a collection of ports representing the local and remote ports that are known to be connected. This approach is appropriate for ATA device and when the underlying drivers or software is limited to information about remote ports and does not include details of the logical devices connected to remote ports. Figure 16: "Optional Connectivity Collection Model" depicts the optional connectivity collection model.

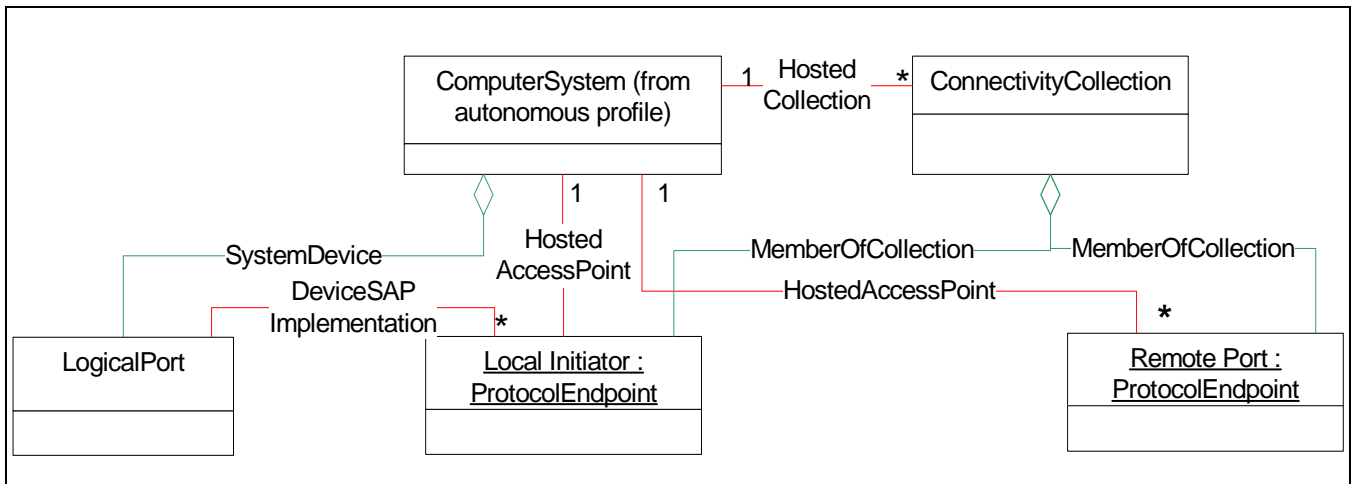


Figure 16 - Optional Connectivity Collection Model

The nature of membership in the collection varies with transports and configuration options. For example, in a parallel SCSI environment, the ConnectivityCollection includes all initiators/targets attached to the bus. In an FC fabric environment, the ConnectivityCollection contains ports that share a zone. In many cases, the ConnectivityCollection could include remote initiators as well as remote devices.

The second approach to modeling remote devices is to include the full initiator/target/logical-unit path model that describes multipath connectivity. This approach has the advantage of including the logical units and including the full path connectivity. The disadvantage is that some OSes handle multipath support in different components from HBA support, making it more efficient to provide the multipath model

as part of the Host Discovered Resources Profile. Figure 17: "Optional Full-Path Model" depicts the optional full-path model.

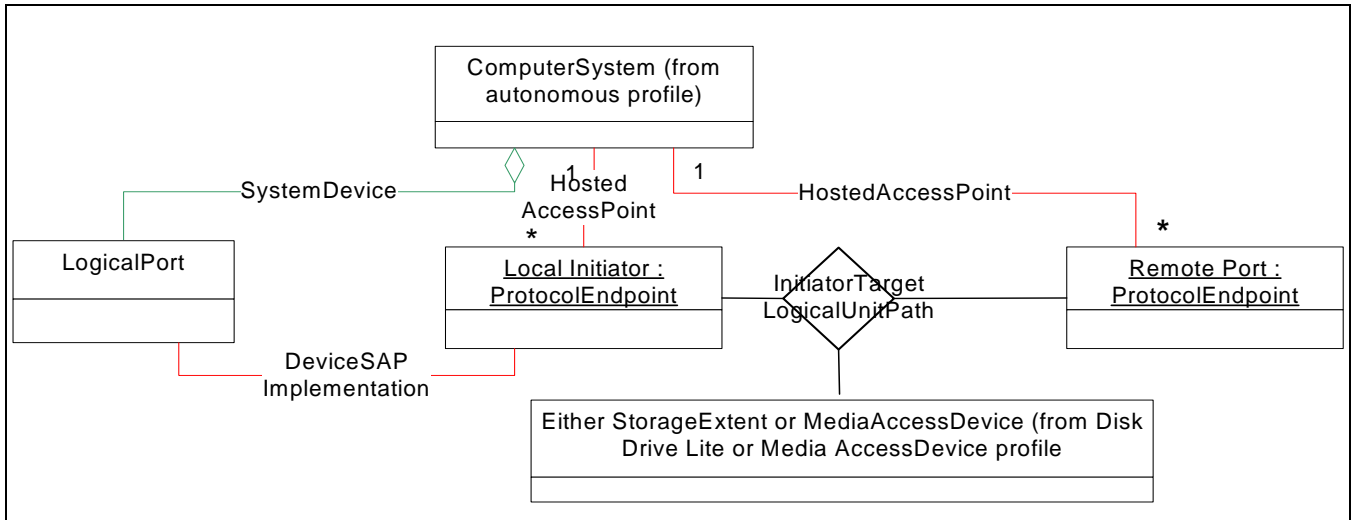


Figure 17 - Optional Full-Path Model

The instrumentation may support the full-path and connectivity collection options by making appropriate ProtocolEndpoints members of ConnectivityCollections.

13.3.1.1 Optional Model for Attached Disks

Disks are modeled using the full-path model and the Disk Drive Lite Profile. The appropriate subclass of InitiatorTargetLogicalUnitPath shall be dependent on whether the disks use the SCSI, ATA or SB commend set. This association references StorageExtent and initiator and target ProtocolEndpoints. The association also provides the disk’s logical unit number. The target ProtocolEndpoint referenced from InitiatorTargetLogicalUnitPath shall be the ProtocolEndpoint from the Disk Drive Lite Profile associated indirectly to StorageExtent via DiskDrive. This is the same ProtocolEndpoint described as the optional remote ProtocolEndpoint in initiator ports profiles.

The ProtocolEndpoints may be associated to a ConnectivityCollection representing a collection of logically connected devices, as illustrated in Figure 18.

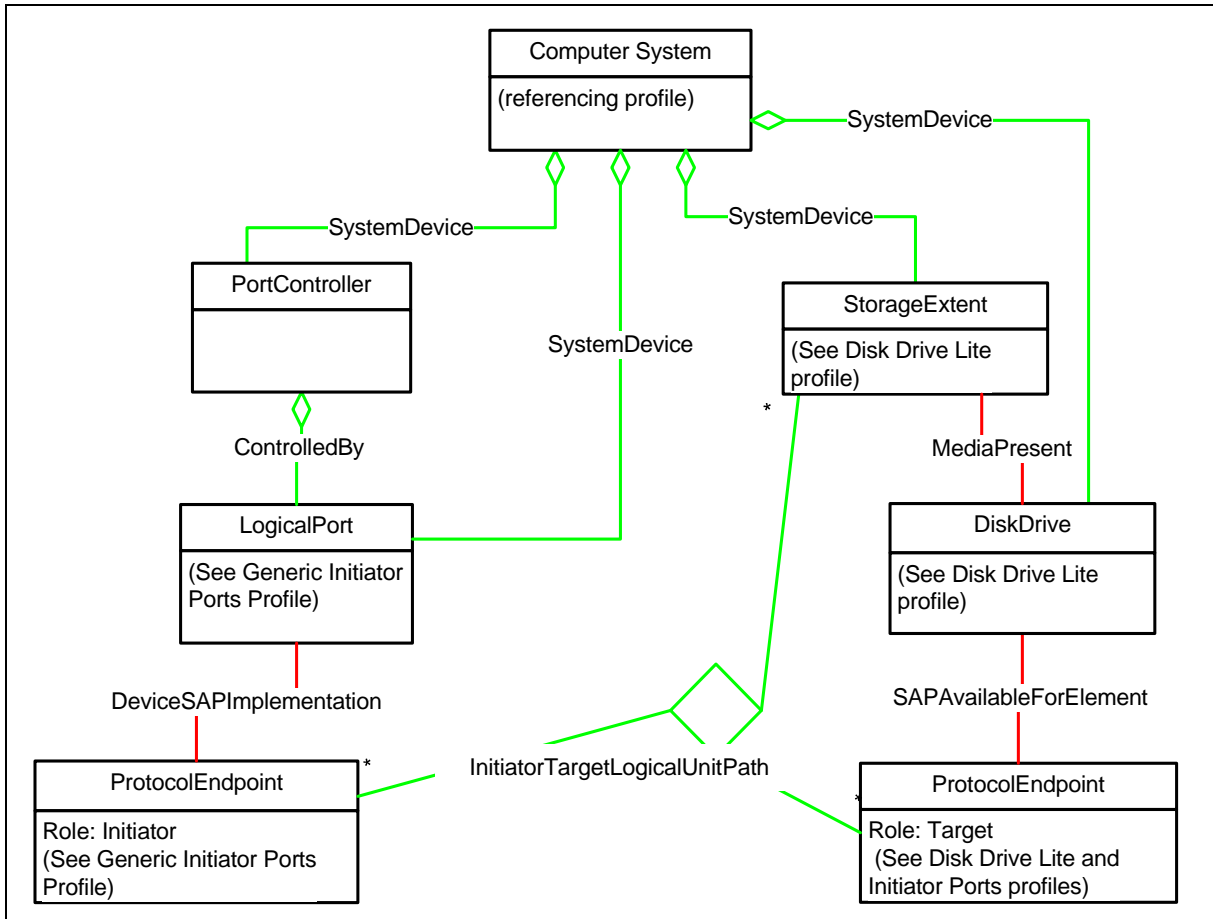


Figure 18 - HBA and Disk Model

13.3.1.2 Optional Model for attached Tape/CD/DVD Drives

The model, illustrated in Figure 19, and requirements are similar to those for disks (see 13.3.1.1), but use the Media Access Device Profile rather than Disk Drive Lite and the appropriate subclass of MediaAccessDevice rather than DiskDrive.

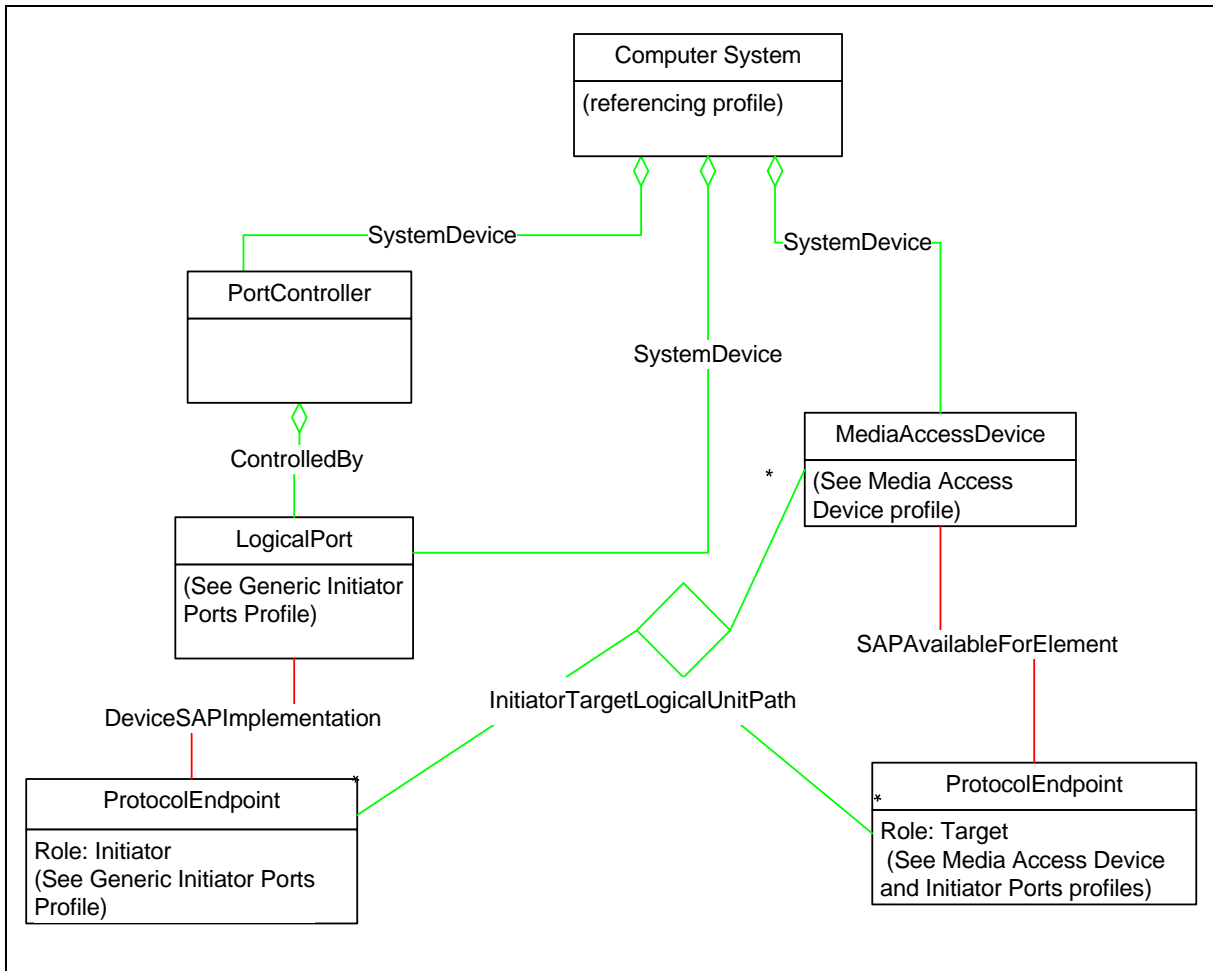


Figure 19 - HBA and Tape or Optical Devices

13.3.1.3 Optional Port Statistics

An implementation of an initiator port profile may optionally support port statistics. Figure 20: "Port Statistics" depicts the model for port statistics.

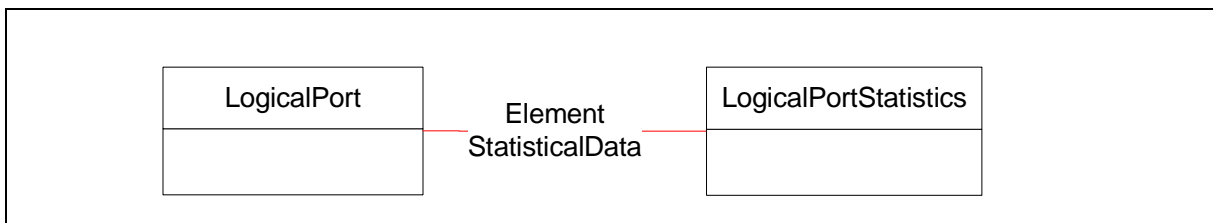


Figure 20 - Port Statistics

If a specialization of the profile specifies a subclass of LogicalPortStatistics (e.g., FCPortStatistics), the implementation should associate that subclass to the appropriate subclass of LogicalPort (e.g., FCPort),

as shown in Figure 21: "Port Statistics Hierarchy". Otherwise, an implementation should use LogicalPortStatistics directly.

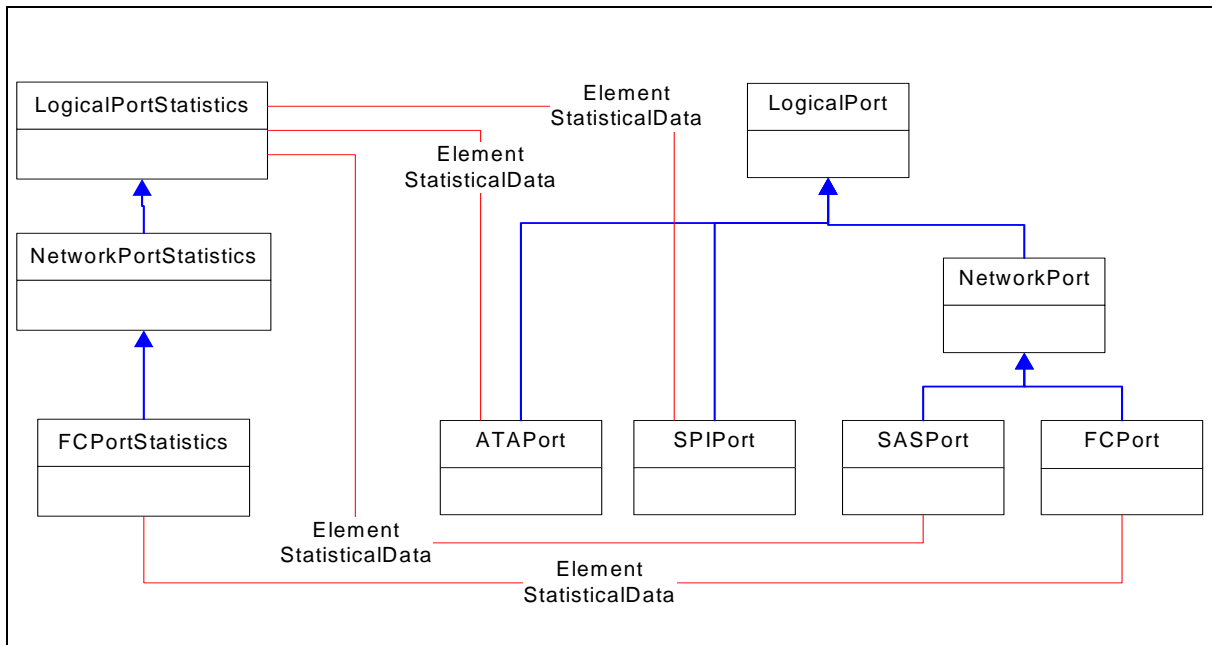


Figure 21 - Port Statistics Hierarchy

13.3.2 Health and Fault Management Considerations

Not defined in this standard.

13.3.3 Cascading Considerations

Not defined in this standard.

13.4 Methods

13.4.1 Extrinsic Methods of this Profile

Not defined in this standard

13.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

13.5 Use Cases

Not defined in this standard.

13.6 CIM Elements

Table 100 describes the CIM elements for Generic Initiator Ports.

Table 100 - CIM Elements for Generic Initiator Ports

Element Name	Requirement	Description
13.6.1 CIM_ConnectivityCollection	Optional	Represents a collection of connected ProtocolEndpoints.
13.6.2 CIM_DeviceSAPImplementation	Mandatory	Connects Initiator LogicalPort and ProtocolEndpoint.
13.6.3 CIM_ElementStatisticalData (Port Statistics)	Optional	Connects LogicalPort and LogicalPortStatistics.
13.6.4 CIM_HostedAccessPoint (Initiator)	Mandatory	Associates system to initiator protocol endpoints.
13.6.5 CIM_HostedAccessPoint (Target)	Optional	Associates system to optional remote protocol endpoints.
13.6.6 CIM_HostedCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates the ConnectivityCollection to the hosting System.
13.6.7 CIM_LogicalPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
13.6.8 CIM_MemberOfCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates ProtocolEndpoints to the ConnectivityCollection.
13.6.9 CIM_ProtocolEndpoint (Initiator)	Mandatory	Represents a protocol (command set) supported by the port. The appropriate subclass (SCSIProtocolEndpoint, ATAProtocolEndpoint, SBProtocolEndpoint) should be used in initiator port specialized profiles.
13.6.10 CIM_ProtocolEndpoint (Target)	Optional	Models protocols of remote ports - target devices and possibly other initiators.
13.6.11 CIM_SystemDevice (Initiator Ports)	Mandatory	Associates system to initiator ports.
13.6.12 CIM_LogicalPortStatistics	Optional	Statistics for a port.

13.6.1 CIM_ConnectivityCollection

Represents a collection of connected ProtocolEndpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 101 describes class CIM_ConnectivityCollection.

Table 101 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	

13.6.2 CIM_DeviceSAPImplementation

Connects Initiator LogicalPort and ProtocolEndpoint.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 102 describes class CIM_DeviceSAPImplementation.

Table 102 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ProtocolEndpoint(Initiator).
Antecedent		Mandatory	Reference to LogicalPort.

13.6.3 CIM_ElementStatisticalData (Port Statistics)

Connects LogicalPort and LogicalPortStatistics.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 103 describes class CIM_ElementStatisticalData (Port Statistics).

Table 103 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to LogicalPort.
Stats		Mandatory	Reference to LogicalPortStatistics.

13.6.4 CIM_HostedAccessPoint (Initiator)

Associates system to initiator protocol endpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 104 describes class CIM_HostedAccessPoint (Initiator).

Table 104 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to ProtocolEndpoint(Initiator).

13.6.5 CIM_HostedAccessPoint (Target)

Associates system to optional remote protocol endpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 105 describes class CIM_HostedAccessPoint (Target).

Table 105 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to ProtocolEndpoint(Target).

13.6.6 CIM_HostedCollection (Connectivity Collection)

Associates the ConnectivityCollection to the hosting System.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 106 describes class CIM_HostedCollection (Connectivity Collection).

Table 106 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ConnectivityCollection.
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.

13.6.7 CIM_LogicalPort

Represents the logical aspects of the physical port and may have multiple associated protocols.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 107 describes class CIM_LogicalPort.

Table 107 - SMI Referenced Properties/Methods for CIM_LogicalPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 3 for ports restricted to back-end (initiator) only or 4 if the port is unrestricted.
PortType		Mandatory	Initiator port specialized profiles specify the appropriate subset of values.

13.6.8 CIM_MemberOfCollection (Connectivity Collection)

Associates ProtocolEndpoints to the ConnectivityCollection.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 108 describes class CIM_MemberOfCollection (Connectivity Collection).

Table 108 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Member		Mandatory	Reference to ProtocolEndpoint.
Collection		Mandatory	Reference to ConnectivityCollection.

13.6.9 CIM_ProtocolEndpoint (Initiator)

Represents a protocol (command set) supported by the port.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 109 describes class CIM_ProtocolEndpoint (Initiator).

Table 109 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	

Table 109 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	
Name	C	Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 7.6.3 Standard Formats for Port Names</i> .
ProtocollFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI', 'ATA', or 'SB'. Initiator port specialized profiles specify the appropriate subset.

13.6.10 CIM_ProtocolEndpoint (Target)

Models protocols of remote ports - target devices and possibly other initiators. The appropriate subclass (SCSIProtocolEndpoint, ATAProtocolEndpoint, SBProtocolEndpoint) should be used in initiator port specialized profiles.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 110 describes class CIM_ProtocolEndpoint (Target).

Table 110 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocollFType		Mandatory	The values in MOFs map to IETF values and exclude storage. Shall be 1 (Other) and set OtherTypeDescription appropriately.
OtherTypeDescription		Mandatory	Shall be the string 'SCSI', 'ATA', or 'SB'. Initiator port specialized profiles specify the appropriate subset.

13.6.11 CIM_SystemDevice (Initiator Ports)

Associates system to initiator ports.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 111 describes class CIM_SystemDevice (Initiator Ports).

Table 111 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to LogicalPort.

13.6.12 CIM_LogicalPortStatistics

Statistics for a port.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 112 describes class CIM_LogicalPortStatistics.

Table 112 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
InstanceID		Mandatory	
BytesTransmitted		Mandatory	
BytesReceived		Mandatory	
PacketsTransmitted		Mandatory	
PacketsReceived		Mandatory	

IMPLEMENTED

DEPRECATED

14 Parallel SCSI (SPI) Initiator Ports Profile

14.1 Synopsis

Profile Name: SPI Initiator Ports (Component Profile)

Version: 1.4.0

Organization: SNIA

Specializes: Generic Initiator Port Profile

Central Class: SPIPort

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

The SPI Initiator Ports Profile models the behavior of a parallel SCSI (SPI) initiator port.

14.2 Description

The SPI Initiator Port Profile defines the model to parallel SCSI ports.

14.3 Implementation

A typical instance diagram is provided in Figure 22: "SPI Initiator Port Instance Diagram".

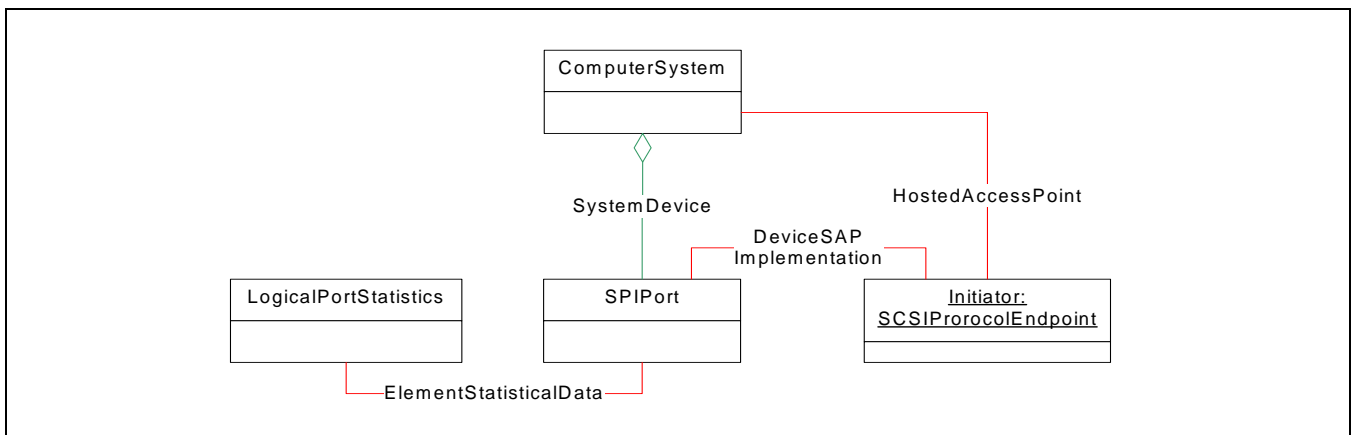


Figure 22 - SPI Initiator Port Instance Diagram

14.3.1 Health and Fault Management Considerations

Table 113 summarizes the Health and Fault Management issues that are unique to this profile.

Table 113 - SPIPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled

Table 113 - SPIPort OperationalStatus

OperationalStatus	Description
InService	Port is in Self Test
Unknown	

14.3.2 Cascading Considerations

Not defined in this standard.

14.4 Methods

14.4.1 Extrinsic Methods of this Profile

Not defined in this standard

14.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported

are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

14.5 Use Cases and Recipes

No recipes are defined in this version of the standard.

14.6 CIM Elements

Table 114 describes the CIM elements for SPI Initiator Ports.

Table 114 - CIM Elements for SPI Initiator Ports

Element Name	Requirement	Description
14.6.1 CIM_ConnectivityCollection	Optional	Represents a collection of connected SCSIProtocolEndpoints.
14.6.2 CIM_DeviceSAPImplementation	Mandatory	Connects Initiator SPILogicalPort and SCSIProtocolEndpoint.
14.6.3 CIM_ElementStatisticalData (Port Statistics)	Optional	Connects SPIPort and LogicalPortStatistics.
14.6.4 CIM_HostedAccessPoint (Initiator)	Mandatory	Associates system to initiator protocol endpoints.
14.6.5 CIM_HostedAccessPoint (Target)	Optional	Associates system to optional remote protocol endpoints.

Table 114 - CIM Elements for SPI Initiator Ports

Element Name	Requirement	Description
14.6.6 CIM_HostedCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates the ConnectivityCollection to the hosting System.
14.6.7 CIM_MemberOfCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Represents a collection of connected SCSIProtocolEndpoints.
14.6.8 CIM_SCSIInitiatorTargetLogicalUnitPath	Optional	Represents a path between a SCSI initiator, target, and logical unit.
14.6.9 CIM_SCSIProtocolEndpoint (Initiator)	Mandatory	Represents support for the SCSI command set.
14.6.10 CIM_SCSIProtocolEndpoint (Target)	Optional	Models protocols of remote ports - target devices and possibly other initiators.
14.6.11 CIM_SPIPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
14.6.12 CIM_SystemDevice (Initiator Ports)	Mandatory	Associates system to initiator ports.
14.6.13 CIM_LogicalPortStatistics	Optional	Statistics for a port.

14.6.1 CIM_ConnectivityCollection

Represents a collection of connected SCSIProtocolEndpoints. The class definition specializes the CIM_ConnectivityCollection definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 115 describes class CIM_ConnectivityCollection.

Table 115 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	

14.6.2 CIM_DeviceSAPImplementation

Connects Initiator SPILogicalPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 116 describes class CIM_DeviceSAPImplementation.

Table 116 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).
Antecedent		Mandatory	Reference to SPIPort.

14.6.3 CIM_ElementStatisticalData (Port Statistics)

Connects SPIPort and LogicalPortStatistics. The class definition specializes the CIM_ElementStatisticalData definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 117 describes class CIM_ElementStatisticalData (Port Statistics).

Table 117 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to SPIPort.
Stats		Mandatory	Reference to LogicalPortStatistics.

14.6.4 CIM_HostedAccessPoint (Initiator)

Associates system to initiator protocol endpoints. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 118 describes class CIM_HostedAccessPoint (Initiator).

Table 118 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).

14.6.5 CIM_HostedAccessPoint (Target)

Associates system to optional remote protocol endpoints. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static
Requirement: Optional

Table 119 describes class CIM_HostedAccessPoint (Target).

Table 119 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Target).

14.6.6 CIM_HostedCollection (Connectivity Collection)

Associates the ConnectivityCollection to the hosting System.

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 120 describes class CIM_HostedCollection (Connectivity Collection).

Table 120 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ConnectivityCollection.
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.

14.6.7 CIM_MemberOfCollection (Connectivity Collection)

Represents a collection of connected SCSIProtocolEndpoints. The class definition specializes the CIM_MemberOfCollection definition in the Generic Initiator Ports profile.

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 121 describes class CIM_MemberOfCollection (Connectivity Collection).

Table 121 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Member		Mandatory	Reference to SCSIProtocolEndpoint(Initiator or Target).
Collection		Mandatory	Reference to ConnectivityCollection.

14.6.8 CIM_SCSIInitiatorTargetLogicalUnitPath

Represents a path between a SCSI initiator, target, and logical unit.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 122 describes class CIM_SCSIInitiatorTargetLogicalUnitPath.

Table 122 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath

Properties	Flags	Requirement	Description & Notes
LogicalUnit		Mandatory	Reference to StorageExtent in Disk Drive Lite Profile or MediaAccessDevice in Media Access Device Profile.
Target		Mandatory	Reference to SCSIProtocolEndpoint(Target).
Initiator		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).

14.6.9 CIM_SCSIProtocolEndpoint (Initiator)

Represents support for the SCSI command set. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 123 describes class CIM_SCSIProtocolEndpoint (Initiator).

Table 123 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	C	Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 7.6.3 Standard Formats for Port Names.</i>
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI', 'ATA', or 'SB'. Initiator port specialized profiles specify the appropriate subset.
ConnectionType		Mandatory	Shall be 3 (Parallel SCSI).
Role		Mandatory	Shall be 2 (Initiator).

14.6.10 CIM_SCSIProtocolEndpoint (Target)

Models protocols of remote ports - target devices and possibly other initiators. The appropriate subclass (SCSIProtocolEndpoint, ATAProtocolEndpoint, SBProtocolEndpoint) should be used in initiator port specialized profiles. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 124 describes class CIM_SCSIProtocolEndpoint (Target).

Table 124 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	The values in MOFs map to IETF values and exclude storage. Shall be 1 (Other) and set OtherTypeDescription appropriately.
OtherTypeDescription		Mandatory	Shall be the string 'SCSI', 'ATA', or 'SB'. Initiator port specialized profiles specify the appropriate subset.
ConnectionType		Mandatory	Shall be 3 (Parallel SCSI).

14.6.11 CIM_SPIPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 125 describes class CIM_SPIPort.

Table 125 - SMI Referenced Properties/Methods for CIM_SPIPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	Shall be 0 (Unknown), 2 (OK), 6 (Error), 10 (Stopped), or 11 (In Service).
UsageRestriction		Mandatory	Shall be 3 (Back-end Only).
PortType		Mandatory	Shall be 140 (SCSI Parallel).

14.6.12 CIM_SystemDevice (Initiator Ports)

Associates system to initiator ports. The class definition specializes the CIM_SystemDevice definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 126 describes class CIM_SystemDevice (Initiator Ports).

Table 126 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to SPIPort(Initiator).

14.6.13 CIM_LogicalPortStatistics

Statistics for a port.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 127 describes class CIM_LogicalPortStatistics.

Table 127 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
InstanceID		Mandatory	
BytesTransmitted		Mandatory	
BytesReceived		Mandatory	
PacketsTransmitted		Mandatory	
PacketsReceived		Mandatory	

DEPRECATED

Parallel SCSI (SPI) Initiator Ports Profile

Parallel SCSI (SPI) Initiator Ports Profile

IMPLEMENTED

15 iSCSI Initiator Port Profile

15.1 Synopsis

Profile Name: iSCSI Initiator Ports (Component Profile)

Version: 1.2.0

Organization: SNIA

Specializes: Generic Initiator Port Profile

Central Class: CIM_EthernetPort

Scoping Class: a CIM_System in a separate autonomous profile

Related Profiles: Table 128 describes the related profiles for iSCSI Initiator Ports.

Table 128 - Related Profiles for iSCSI Initiator Ports

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

Models an adapter (NIC, HBA, TOE) for iSCSI.

15.2 Description

Models an adapter (NIC, HBA, TOE) for iSCSI.

15.3 Implementation

Other port profiles have a single physical port (LogicalPort subclass) associated with each SCSI initiator (SCSIProtocolEndpoint). iSCSI allows multiple connections (each with a single Ethernet port) in a session that acts as a SCSI initiator. This profile includes the subset of classes that model the SCSI initiator and its relationship to logical classes that model physical elements (Ethernet ports).

Figure 23 depicts a configuration with an initiator with two Ethernet ports that are part of a single session that acts as a SCSI initiator. The Ethernet ports (referred to in iSCSI literature as Network Portals) are modeled as instances of EthernetPort, IPProtocolEndpoint, and TCPProtocolEndpoint with 1-1 cardinality. These ports are in the initiator side, the target ports are not required in this profile. Note that all ProtocolEndpoint instances need a HostAccessPoint association to the ComputerSystem, some are omitted to keep the diagram less cluttered.

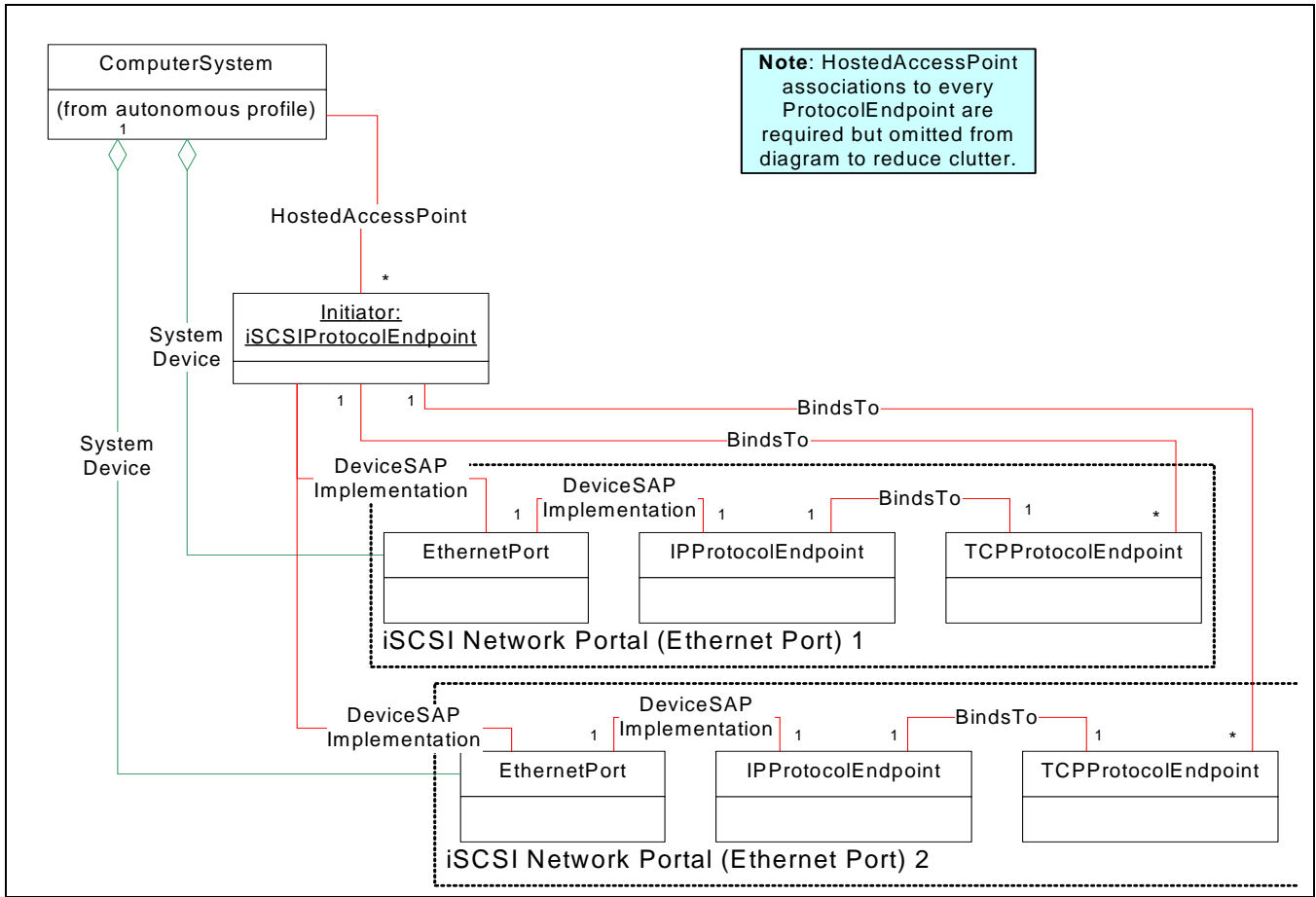


Figure 23 - iSCSI Initiator Port Instance Diagram

15.3.1 Health and Fault Management Considerations

Table 129 describes EthernetPort OperationalStatus.

Table 129 - EthernetPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

15.3.2 Cascading Considerations

Not defined in this standard.

15.4 Methods

15.4.1 Extrinsic Methods of this Profile

Not defined in this standard

15.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

15.5 Use Cases and Recipes

No recipes are defined in this version of the standard.

15.6 CIM Elements

Table 130 describes the CIM elements for iSCSI Initiator Ports.

Table 130 - CIM Elements for iSCSI Initiator Ports

Element Name	Requirement	Description
15.6.1 CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)	Mandatory	
15.6.2 CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint)	Mandatory	
15.6.3 CIM_DeviceSAPImplementation (IPProtocolEndpoint to EthernetPort)	Mandatory	
15.6.4 CIM_DeviceSAPImplementation (iSSIProtocolEndpoint to EthernetPort)	Mandatory	
15.6.5 CIM_EthernetPort	Mandatory	
15.6.6 CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint)	Mandatory	
15.6.7 CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint)	Mandatory	
15.6.8 CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint)	Mandatory	
15.6.9 CIM_IPProtocolEndpoint	Mandatory	
15.6.10 CIM_iSCSIProtocolEndpoint	Mandatory	
15.6.11 CIM_SystemDevice (ComputerSystem to EthernetPort)	Mandatory	

Table 130 - CIM Elements for iSCSI Initiator Ports

Element Name	Requirement	Description
15.6.12 CIM_TCPProtocolEndpoint	Mandatory	
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_EthernetPort	Mandatory	Port Creation.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_EthernetPort AND SourceInstance.CIM_EthernetPort::OperationalStatus <> PreviousInstance.CIM_EthernetPort::OperationalStatus	Mandatory	Port Status Change.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_EthernetPort	Mandatory	Port Removal.

15.6.1 CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 131 describes class CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint) .

Table 131 - SMI Referenced Properties/Methods for CIM_BindsTo (iSCSIProtocolEndpoint to TCPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to TCPProtocolEndpoint
Dependent		Mandatory	Reference to iSCSIProtocolEndpoint

15.6.2 CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 131 describes class CIM_BindsTo (TCPProtocolEndpoint to IPProtocolEndpoint).

Table 132 - SMI Referenced Properties/Methods for CIM_BindsTo(TCPProtocolEndpoint to IPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to IPProtocolEndpoint
Dependent		Mandatory	Reference to TCPProtocolEndpoint

15.6.3 CIM_DeviceSAPImplementation (IPProtocolEndpoint to EthernetPort)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 133 describes class CIM_DeviceSAPImplementation (IPProtocolEndpoint to EthernetPort).

Table 133 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (IPProtocolEndpoint to EthernetPort)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to EthernetPort
Dependent		Mandatory	Reference to IPProtocolEndpoint

15.6.4 CIM_DeviceSAPImplementation (iSSIPProtocolEndpoint to EthernetPort)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 134 describes class CIM_DeviceSAPImplementation (iSSIPProtocolEndpoint to EthernetPort).

Table 134 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation (iSCSIProtocolEndpoint to EthernetPort)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to EthernetPort
Dependent		Mandatory	Reference to iSCSIProtocolEndpoint

15.6.5 CIM_EthernetPort

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 135 describes class CIM_EthernetPort .

Table 135 - SMI Referenced Properties/Methods for CIM_EthernetPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
PortType		Mandatory	
OperationalStatus		Mandatory	
PermanentAddress	CD	Mandatory	

15.6.6 CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 136 describes class CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint).

Table 136 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to IPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to IPProtocolEndpoint

15.6.7 CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 137 describes class CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint).

Table 137 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to TCPProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to TCPProtocolEndpoint

15.6.8 CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 138 describes class CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint).

Table 138 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (ComputerSystem to iSCSIProtocolEndpoint)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to iSCSIProtocolEndpoint

15.6.9 CIM_IPProtocolEndpoint

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 139 describes class CIM_IPProtocolEndpoint .

Table 139 - SMI Referenced Properties/Methods for CIM_IPProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
IPv4Address	CD	Optional	Maps to IMA_NETWORK_PORTAL_PROPERTIES, ipAddress.
IPv6Address	CD	Optional	Maps to IMA_NETWORK_PORTAL_PROPERTIES, ipAddress.
ProtocolIFType		Mandatory	

15.6.10 CIM_iSCSIProtocolEndpoint

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 140 describes class CIM_iSCSIProtocolEndpoint .

Table 140 - SMI Referenced Properties/Methods for CIM_iSCSIProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	CD	Mandatory	MUST contain the iSCSI port name as described in the iSCSI RFC (currently http://www.ietf.org/rfc/rfc3720.txt) - < iSCSI node name> + ',i,' + ISID for initiators
ProtocolIFType		Mandatory	Shall be Other (1).
OtherTypeDescription		Mandatory	Shall be "iSCSI"
ConnectionType		Mandatory	iSCSI.
Role		Mandatory	Shall be 2 (Initiator).
Identifier		Mandatory	ISID.

15.6.11 CIM_SystemDevice (ComputerSystem to EthernetPort)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 141 describes class CIM_SystemDevice (ComputerSystem to EthernetPort).

Table 141 - SMI Referenced Properties/Methods for CIM_SystemDevice (ComputerSystem to EthernetPort)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem
PartComponent		Mandatory	Reference to EthernetPort

15.6.12 CIM_TCPProtocolEndpoint

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 142 describes class CIM_TCPProtocolEndpoint .

Table 142 - SMI Referenced Properties/Methods for CIM_TCPProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
PortNumber	CD	Mandatory	
ProtocolType		Mandatory	

IMPLEMENTED

STABLE**16 FC Initiator Ports Profile****16.1 Synopsis****Profile Name:** FC Initiator Ports (Component Profile)**Version:** 1.7.0**Organization:** SNIA**Specializes:** Generic Initiator Ports Profile**Central Class:** FCPort**Scoping Class:** a ComputerSystem in a referencing autonomous profile**Related Profiles:** Table 143 describes the related profiles for FC Initiator Ports.**Table 143 - Related Profiles for FC Initiator Ports**

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

The FC Initiator Ports Profile models the behavior of a Fibre Channel port supporting FCP (SCSI command protocol).

16.2 Description

The FC Initiator Ports Profile models the behavior of a Fibre Channel port supporting FCP (SCSI command protocol).

16.3 Implementation

Figure 24: "Fibre Channel Initiator Instance Diagram" is an example of a single port and drive connected to a single system using Fibre Channel. This instance diagram shows a disk (LogicalDevice in the diagram would be subclassed as something like StorageExtent) in an array, connected by a Fibre

Channel port. The full model for the disk is shown in *Storage Management Technical Specification, Part 4 Block Devices, 1.7.0 Rev 5 10 Disk Drive Lite Profile*.

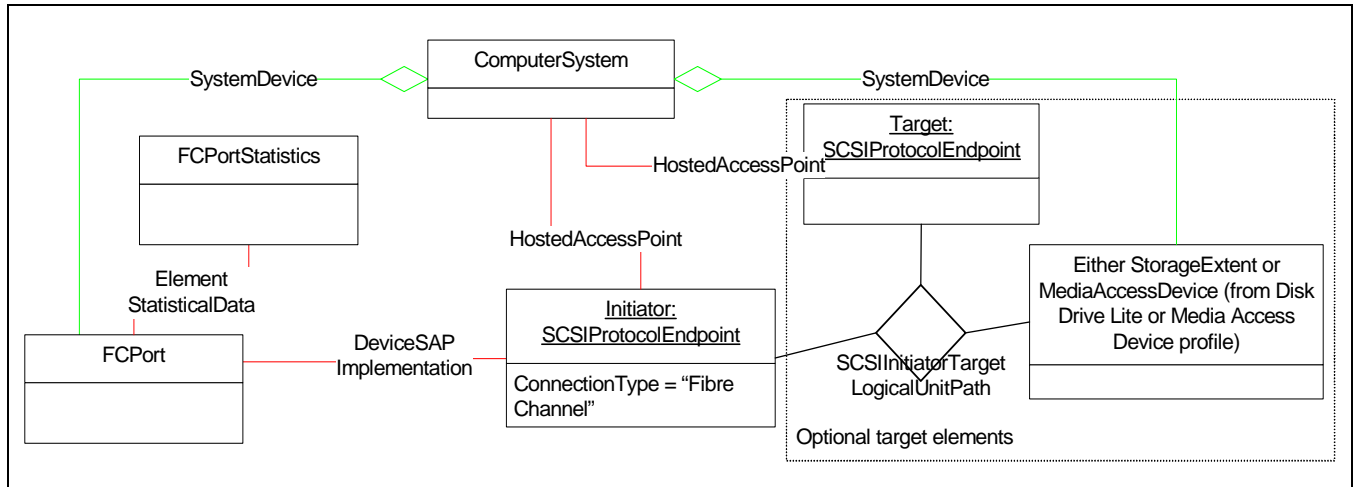


Figure 24 - Fibre Channel Initiator Instance Diagram

16.3.1 Port Statistics

The FCPortStatistics subclass of NetworkPortStatistics is optional. If supported, FCPortStatistics shall be associated to FcPort using ElementStatisticalData.

16.3.2 Logical Port Group (FC Node)

LogicalPortGroup may optionally be used to model the collection of ports that shared a Node WWN (in this case, both ports on a card, but other implementations are in use). If LogicalPortGroup is instantiated, it shall be associated to the ComputerSystem in the referencing profile using HostedCollection and also associated to FCPorts using MemberOfCollection. Figure 25, "FC Node Model" shows to model for FC Nodes.

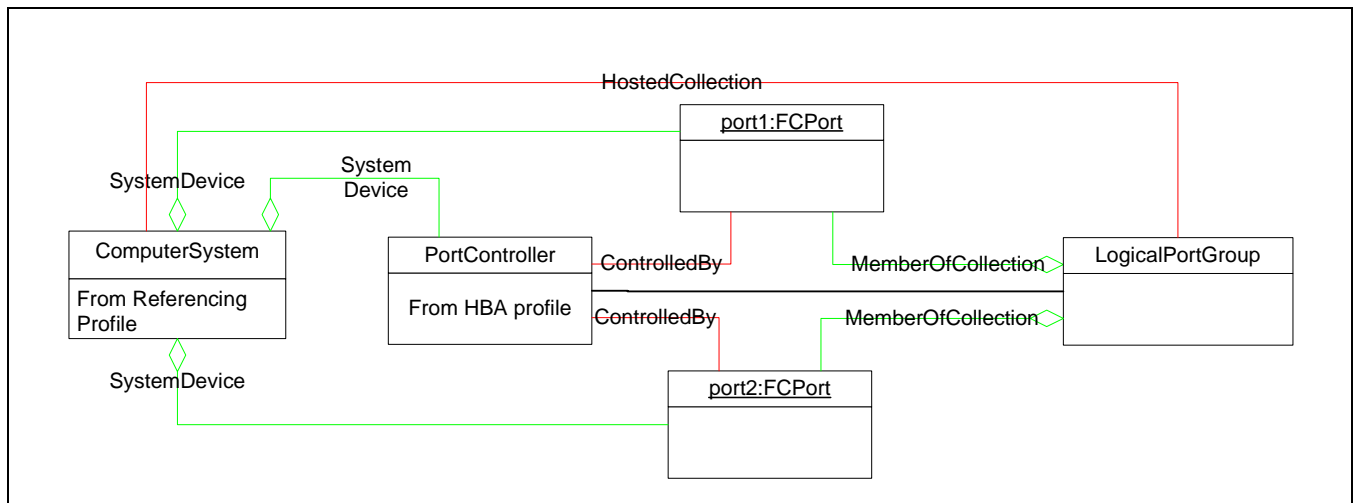


Figure 25 - FC Node Model

16.3.3 Health and Fault Management Considerations

Table 144 summarized the Health and Fault Management considerations specific to this profile.

Table 144 - FCPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

16.3.4 Cascading Considerations

Not defined in this standard.

16.4 Methods

16.4.1 Extrinsic Methods of this Profile

Not defined in this standard

16.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

16.5 Use Cases and Recipes

No recipes are defined in this version of the standard.

16.6 CIM Elements

Table 145 describes the CIM elements for FC Initiator Ports.

Table 145 - CIM Elements for FC Initiator Ports

Element Name	Requirement	Description
16.6.1 CIM_ConnectivityCollection	Optional	Represents a collection of connected ProtocolEndpoints.
16.6.2 CIM_DeviceSAPImplementation	Mandatory	Connects Initiator FCPort and SCSIProtocolEndpoint.
16.6.3 CIM_ElementStatisticalData (Port Statistics)	Conditional	Conditional requirement: support for the FC HBA profile. Connects FCPort and FCPortStatistics.

Table 145 - CIM Elements for FC Initiator Ports

Element Name	Requirement	Description
16.6.4 CIM_FCPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
16.6.5 CIM_FCPortStatistics	Conditional	Conditional requirement: support for the FC HBA profile. Statistics for a port.
16.6.6 CIM_HostedAccessPoint (Initiator)	Mandatory	Associates system to initiator protocol endpoints.
16.6.7 CIM_HostedAccessPoint (Target)	Optional	Associates system to optional remote protocol endpoints.
16.6.8 CIM_HostedCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates the ConnectivityCollection to the hosting System.
16.6.9 CIM_MemberOfCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates SCSIProtocolEndpoints to the ConnectivityCollection.
16.6.10 CIM_SCSIInitiatorTargetLogicalUnitPath	Optional	Represents a path between a SCSI initiator, target, and logical unit.
16.6.11 CIM_SCSIProtocolEndpoint (Initiator)	Mandatory	Represents support for the SCSI command set.
16.6.12 CIM_SCSIProtocolEndpoint (Target)	Optional	Models remote ports - target devices and possibly other initiators.
16.6.13 CIM_SystemDevice (Initiator Ports)	Mandatory	Associates system to initiator ports.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_FCPort	Optional	Create FCPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::OperationalStatus <>PreviousInstance.CIM_FCPort::OperationalStatus	Optional	CQL -Modify FCPort.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FCPort	Optional	Delete FCPort.

16.6.1 CIM_ConnectivityCollection

Represents a collection of connected ProtocolEndpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 146 describes class CIM_ConnectivityCollection.

Table 146 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	

16.6.2 CIM_DeviceSAPImplementation

Connects Initiator FCPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 147 describes class CIM_DeviceSAPImplementation.

Table 147 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to Initiator SCSIProtocolEndpoint.
Antecedent		Mandatory	Reference to FCPort.

16.6.3 CIM_ElementStatisticalData (Port Statistics)

Connects FCPort and FCPortStatistics. The class definition specializes the CIM_ElementStatisticalData definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: support for the FC HBA profile.

Table 148 describes class CIM_ElementStatisticalData (Port Statistics).

Table 148 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to FCPort.
Stats		Mandatory	Reference to FCPortStatistics.

16.6.4 CIM_FCPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 149 describes class CIM_FCPort.

Table 149 - SMI Referenced Properties/Methods for CIM_FCPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	Shall be 0 (Unknown), 2 (OK), 6 (Error), 10 (Stopped), or 11 (In Service).
UsageRestriction		Mandatory	Shall be 3 for ports restricted to back-end (initiator) only or 4 if the port is unrestricted.
PortType		Mandatory	Shall be 0 1 10 11 12 13 14 15 16 17 18 (Unknown or Other or N or NL or F/NL or Nx or E or F or FL or B or G).
OtherPortType		Conditional	Required when PortType is Other
ElementName		Mandatory	Port Symbolic Name.
Speed		Mandatory	Speed in bits per second. Shall be 0, 1062500000 (1GFC), 2125000000 (2GFC), 4250000000 (4GFC), 8500000000 (8GFC), 10518750000 (10GFC), 14025000000 (16GFC), 21037500000 (20GFC) or 28500000000 (32GFC).
MaxSpeed		Mandatory	Maximum Port Speed.
PortNumber		Optional	
PermanentAddress	CD	Optional	Port WWN. PermanentAddress is optional when used as a back-end port in a device. This may be overridden in profiles that use this profile. Shall be 16 un-separated upper case hex digits. See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 7.6.3 Standard Formats for Port Names</i> .
NetworkAddresses		Optional	For Fibre Channel end device ports, the Fibre Channel ID. Shall be 16 un-separated upper case hex digits.
SupportedCOS		Optional	Shall be 0 (unknown), 1 (Class 1), 2 (Class 2), 3, (Class 3), 4 (Class 4), 6 (Class 6), or 7 (Class 7).
ActiveCOS		Optional	Shall be 0 (unknown), 1 (Class 1), 2 (Class 2), 3, (Class 3), 4 (Class 4), 6 (Class 6), or 7 (Class 7).
LinkTechnology		Mandatory	Shall be 4 (FC).
SupportedMaximumTransmissionUnit		Mandatory	
PortDiscriminator		Conditional	Experimental. Conditional requirement: support for the Storage HBA profile. Shall include 11 (FC Native) and 12 (HBA).

16.6.5 CIM_FCPortStatistics

Statistics for a port. The class definition specializes the CIM_LogicalPortStatistics definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: support for the FC HBA profile.

Table 150 describes class CIM_FCPortStatistics.

Table 150 - SMI Referenced Properties/Methods for CIM_FCPortStatistics

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
InstanceID		Mandatory	
BytesTransmitted		Mandatory	From NetworkPortStatistics Superclass. Maps to HBA_PortStatistics.TxWords. Multiply word count by 4.
BytesReceived		Mandatory	From NetworkPortStatistics Superclass. Maps to HBA_PortStatistics.RxWords. Multiply word count by 4.
PacketsTransmitted		Mandatory	From NetworkPortStatistics Superclass. Maps to HBA_PortStatistics.TxFrames.
PacketsReceived		Mandatory	From NetworkPortStatistics Superclass. Maps to HBA_PortStatistics.RxFrames.
CRCErrors		Mandatory	Maps to HBA_PortStatistics.InvalidCRCCCount.
LinkFailures		Mandatory	Maps to HBA_PortStatistics.LinkFailureCount.
PrimitiveSeqProtocolErrCount		Mandatory	
LossOfSignalCounter		Mandatory	Maps to HBA_PortStatistics.LossOfSignalCount.
InvalidTransmissionWords		Mandatory	Maps to HBA_PortStatistics.InvalidTxWordCount.
StatisticTime		Optional	Time last measurement was taken.
LIPCount		Mandatory	
NOSCount		Mandatory	
ErrorFrames		Mandatory	
DumpedFrames		Mandatory	
LossOfSyncCounter		Mandatory	Maps to HBA_PortStatistics.LossOfSynchCount.

16.6.6 CIM_HostedAccessPoint (Initiator)

Associates system to initiator protocol endpoints. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 151 describes class CIM_HostedAccessPoint (Initiator).

Table 151 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).

16.6.7 CIM_HostedAccessPoint (Target)

Associates system to optional remote protocol endpoints. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 152 describes class CIM_HostedAccessPoint (Target).

Table 152 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Target).

16.6.8 CIM_HostedCollection (Connectivity Collection)

Associates the ConnectivityCollection to the hosting System.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 153 describes class CIM_HostedCollection (Connectivity Collection).

Table 153 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ConnectivityCollection.
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.

16.6.9 CIM_MemberOfCollection (Connectivity Collection)

Associates SCSIProtocolEndpoints to the ConnectivityCollection. The class definition specializes the CIM_MemberOfCollection definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 154 describes class CIM_MemberOfCollection (Connectivity Collection).

Table 154 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Member		Mandatory	Reference to target or initiator SCSIProtocolEndpoint.
Collection		Mandatory	Reference to ConnectivityCollection.

16.6.10 CIM_SCSIInitiatorTargetLogicalUnitPath

Represents a path between a SCSI initiator, target, and logical unit.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 155 describes class CIM_SCSIInitiatorTargetLogicalUnitPath.

Table 155 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath

Properties	Flags	Requirement	Description & Notes
LogicalUnit		Mandatory	Reference to StorageExtent in Disk Drive Lite Profile or MediaAccessDevice in Media Access Device Profile.
Initiator		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).
Target		Mandatory	Reference to SCSIProtocolEndpoint(Target).

16.6.11 CIM_SCSIProtocolEndpoint (Initiator)

Represents support for the SCSI command set. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 156 describes class CIM_SCSIProtocolEndpoint (Initiator).

Table 156 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	C	Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 7.6.3 Standard Formats for Port Names.</i>
ProtocolIFType		Mandatory	Shall be 1 (Other).

Table 156 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).
Role		Mandatory	Shall be 2 (Initiator) or 4 (Both Initiator and Target).

16.6.12 CIM_SCSIProtocolEndpoint (Target)

Models remote ports - target devices and possibly other initiators. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 157 describes class CIM_SCSIProtocolEndpoint (Target).

Table 157 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	The values in MOFs map to IETF values and exclude storage. Shall be 1 (Other) and set OtherTypeDescription appropriately.
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
Role		Mandatory	Should be set appropriately by the instrumentation. If not known, use 0 (Unknown).
ConnectionType		Mandatory	Shall be 8 (FC).

16.6.13 CIM_SystemDevice (Initiator Ports)

Associates system to initiator ports. The class definition specializes the CIM_SystemDevice definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 158 describes class CIM_SystemDevice (Initiator Ports).

Table 158 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to FCPort.

STABLE

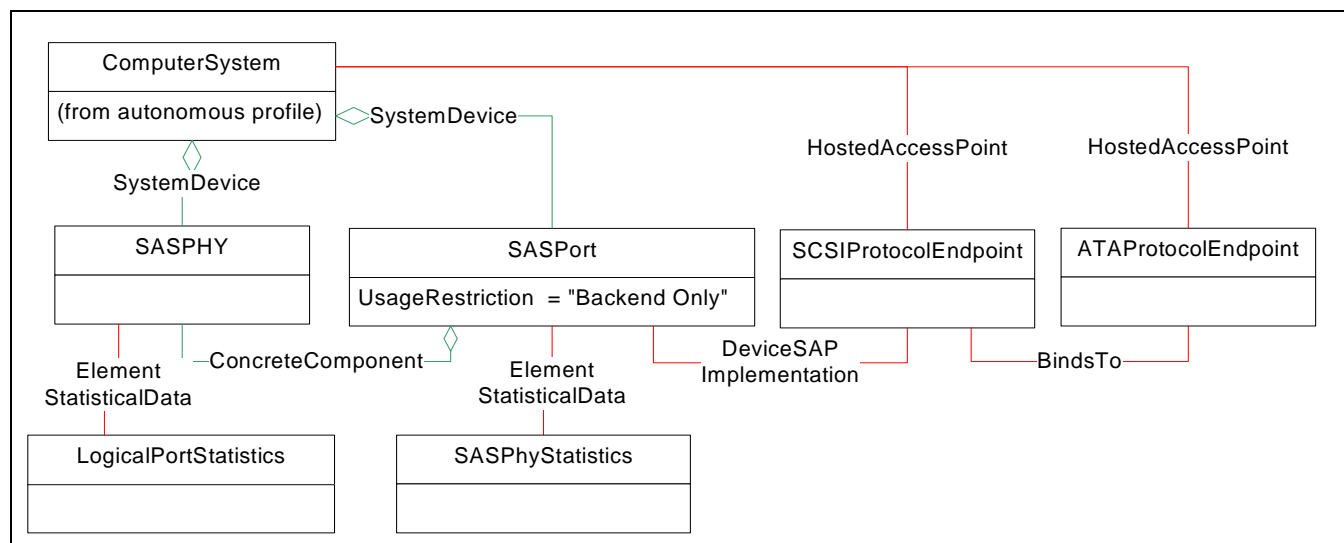
FC Initiator Ports Profile

IMPLEMENTED
17 SAS Initiator Ports Profile**17.1 Synopsis****Profile Name:** SAS Initiator Ports (Component Profile)**Version:** 1.7.0**Organization:** SNIA**Specializes:** Generic Initiator Port Profile**Central Class:** SASPort**Scoping Class:** a ComputerSystem in a separate autonomous profile**Related Profiles:** Not defined in this standard.

The SAS Initiator Port Profile models the management of a Serial Attached SCSI port that initiates commands to devices.

17.2 Description

The SAS Initiator Port Profile defines the model Serial Attached SCSI (SAS) ports. A typical instance diagram is provided in Figure 26: "SAS Initiator Port Model".

**Figure 26 - SAS Initiator Port Model**

The SASPhy class represents a SAS PHY. A SAS Port may have multiple associated PHYs; generally, all the PHYs are connected to the same target or expander and provide additional bandwidth.

SASPort represents a SAS initiator port which is an aggregation of SASPHY instances.

SASPhyStatistics is optional and may be associated to SASPhy to hold PHY error statistics.

LogicalPortStatistics is optional and may be associated to SASPort to hold port I/O statistics.

ATAProtocolEndpoint associated to SCSIProtocolEndpoint using BindsTo shall be used to represent support for ATA (SATA) tunneled over SCSI.

17.2.1 Health and Fault Management Considerations

Table 159 summarizes the Health and Fault Management issues that are unique to this profile.

Table 159 - SASPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

17.3 Methods of the profile

Not defined in this standard.

17.4 Client Considerations and Recipes

Not defined in this standard.

17.5 CIM Elements

Table 160 describes the CIM elements for SAS Initiator Ports.

Table 160 - CIM Elements for SAS Initiator Ports

Element Name	Requirement	Description
17.5.1 CIM_ATAProtocolEndpoint (Initiator)	Optional	Initiator ATA endpoints.
17.5.2 CIM_BindsTo	Optional	Associates SCSIProtocolEndpoint and ATAProtocolEndpoint.
17.5.3 CIM_ConcreteComponent	Mandatory	Associates SASPort and SASPHY.
17.5.4 CIM_ConnectivityCollection	Optional	Represents a collection of connected ProtocolEndpoints.
17.5.5 CIM_DeviceSAPImplementation	Mandatory	Connects Initiator SASLogicalPort and SCSIProtocolEndpoint.
17.5.6 CIM_ElementStatisticalData (PHY Statistics)	Optional	Associates SASPort and SASPhyStatistics.
17.5.7 CIM_ElementStatisticalData (Port Statistics)	Optional	Connects LogicalPort and LogicalPortStatistics.
17.5.8 CIM_HostedAccessPoint (Initiator)	Mandatory	Associates system to initiator protocol endpoints.
17.5.9 CIM_HostedAccessPoint (Target)	Optional	Associates system to optional remote protocol endpoints.
17.5.10 CIM_HostedCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates the ConnectivityCollection to the hosting System.
17.5.11 CIM_MemberOfCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates ProtocolEndpoints to the ConnectivityCollection.

Table 160 - CIM Elements for SAS Initiator Ports

Element Name	Requirement	Description
17.5.12 CIM_SASPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
17.5.13 CIM_SCSIInitiatorTargetLogicalUnitPath	Optional	Represents a path between a SCSI initiator, target, and logical unit.
17.5.14 CIM_SCSIProtocolEndpoint (Initiator)	Mandatory	Represents support for the SCSI command set.
17.5.15 CIM_SCSIProtocolEndpoint (Target)	Optional	Models remote ports - target devices and possibly other initiators.
17.5.16 CIM_SystemDevice (Initiator PHY)	Mandatory	Associates system to initiator SAS PHYs.
17.5.17 CIM_SystemDevice (Initiator Ports)	Mandatory	Associates system to initiator ports.
17.5.18 CIM_LogicalPortStatistics	Optional	Statistics for a port.
17.5.19 CIM_SASPHY	Mandatory	A PHY on a SAS HBA, Expander, or device.
17.5.20 CIM_SASPhyStatistics	Optional	Statistics for a SAS PHY.

17.5.1 CIM_ATAProtocolEndpoint (Initiator)

Initiator ATA endpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 161 describes class CIM_ATAProtocolEndpoint (Initiator).

Table 161 - SMI Referenced Properties/Methods for CIM_ATAProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5</i> 7.6.3 Standard Formats for Port Names.
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be 'ATA'.
ConnectionType		Mandatory	Shall be 3 (SATA).
Role		Mandatory	Shall be 3 (Target).

17.5.2 CIM_BindsTo

Associates SCSIProtocolEndpoint and ATAProtocolEndpoint.

Created By: Static

Modified By: Static

Deleted By: Static
Requirement: Optional

Table 162 describes class CIM_BindsTo.

Table 162 - SMI Referenced Properties/Methods for CIM_BindsTo

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ATAProtocolEndpoint.
Antecedent		Mandatory	Reference to SCSIProtocolEndpoint.

17.5.3 CIM_ConcreteComponent

Associates SASPort and SASPHY.

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Mandatory

Table 163 describes class CIM_ConcreteComponent.

Table 163 - SMI Referenced Properties/Methods for CIM_ConcreteComponent

Properties	Flags	Requirement	Description & Notes
PartComponent		Mandatory	Reference to SASPHY.
GroupComponent		Mandatory	Reference to SASPort.

17.5.4 CIM_ConnectivityCollection

Represents a collection of connected ProtocolEndpoints.

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Optional

Table 164 describes class CIM_ConnectivityCollection.

Table 164 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	

17.5.5 CIM_DeviceSAPImplementation

Connects Initiator SASLogicalPort and SCSIProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 165 describes class CIM_DeviceSAPImplementation.

Table 165 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).
Antecedent		Mandatory	Reference to SASPort.

17.5.6 CIM_ElementStatisticalData (PHY Statistics)

Associates SASPort and SASPhyStatistics.

Requirement: Optional

Table 166 describes class CIM_ElementStatisticalData (PHY Statistics).

Table 166 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (PHY Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Referenceto SASPort.
Stats		Mandatory	Reference to SASPhyStatistics.

17.5.7 CIM_ElementStatisticalData (Port Statistics)

Connects LogicalPort and LogicalPortStatistics.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 167 describes class CIM_ElementStatisticalData (Port Statistics).

Table 167 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to LogicalPort.
Stats		Mandatory	Reference to LogicalPortStatistics.

17.5.8 CIM_HostedAccessPoint (Initiator)

Associates system to initiator protocol endpoints. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static

Deleted By: Static
Requirement: Mandatory

Table 168 describes class CIM_HostedAccessPoint (Initiator).

Table 168 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).

17.5.9 CIM_HostedAccessPoint (Target)

Associates system to optional remote protocol endpoints. The class definition specializes the CIM_HostedAccessPoint definition in the Generic Initiator Ports profile.

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Optional

Table 169 describes class CIM_HostedAccessPoint (Target).

Table 169 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to SCSIProtocolEndpoint(Target).

17.5.10 CIM_HostedCollection (Connectivity Collection)

Associates the ConnectivityCollection to the hosting System.

Created By: Static
Modified By: Static
Deleted By: Static
Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 170 describes class CIM_HostedCollection (Connectivity Collection).

Table 170 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ConnectivityCollection.
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.

17.5.11 CIM_MemberOfCollection (Connectivity Collection)

Associates ProtocolEndpoints to the ConnectivityCollection. The class definition specializes the CIM_MemberOfCollection definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 171 describes class CIM_MemberOfCollection (Connectivity Collection).

Table 171 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Member		Mandatory	Reference to SCSIProtocolEndpoint.
Collection		Mandatory	Reference to ConnectivityCollection.

17.5.12 CIM_SASPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 172 describes class CIM_SASPort.

Table 172 - SMI Referenced Properties/Methods for CIM_SASPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	Shall be 0 (Unknown), 2 (OK), 6 (Error), 10 (Stopped), or 11 (In Service).
UsageRestriction		Mandatory	Shall be 3 (Back-end Only).
PortType		Mandatory	Shall be 94 (SAS).
PermanentAddress		Mandatory	SAS Address. Shall be 16 un-separated upper case hex digits. See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 57.6.3 Standard Formats for Port Names.</i>

17.5.13 CIM_SCSIInitiatorTargetLogicalUnitPath

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 173 describes class CIM_SCSIInitiatorTargetLogicalUnitPath.

Table 173 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath

Properties	Flags	Requirement	Description & Notes
LogicalUnit		Mandatory	Reference to StorageExtent in Disk Drive Lite Profile or MediaAccessDevice in Media Access Device Profile.
Target		Mandatory	Reference to SCSIProtocolEndpoint(Target).
Initiator		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).

17.5.14 CIM_SCSIProtocolEndpoint (Initiator)

Represents support for the SCSI command set. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 174 describes class CIM_SCSIProtocolEndpoint (Initiator).

Table 174 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	C	Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 7.6.3</i> Standard Formats for Port Names.
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 8 (SAS).
Role		Mandatory	Shall be 2 (Initiator).

17.5.15 CIM_SCSIProtocolEndpoint (Target)

Models remote ports - target devices and possibly other initiators. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 175 describes class CIM_SCSIProtocolEndpoint (Target).

Table 175 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	The values in MOFs map to IETF values and exclude storage. Shall be 1 (Other) and set OtherTypeDescription appropriately.
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
Role		Mandatory	SCSI target or initiator role. Should be set appropriately by the instrumentation. If not know, use 0 (Unknown).
ConnectionType		Mandatory	Shall be 8 (SAS).

17.5.16 CIM_SystemDevice (Initiator PHY)

Associates system to initiator SAS PHYs.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 176 describes class CIM_SystemDevice (Initiator PHY).

Table 176 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator PHY)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to SASPHY.

17.5.17 CIM_SystemDevice (Initiator Ports)

Associates system to initiator ports. The class definition specializes the CIM_SystemDevice definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 177 describes class CIM_SystemDevice (Initiator Ports).

Table 177 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to back-end (initiator) SASPorts.

17.5.18 CIM_LogicalPortStatistics

Statistics for a port.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 178 describes class CIM_LogicalPortStatistics.

Table 178 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
InstanceID		Mandatory	
BytesTransmitted		Mandatory	
BytesReceived		Mandatory	
PacketsTransmitted		Mandatory	
PacketsReceived		Mandatory	

17.5.19 CIM_SASPHY

A PHY on a SAS HBA, Expander, or device.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 179 describes class CIM_SASPHY.

Table 179 - SMI Referenced Properties/Methods for CIM_SASPHY

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	

Table 179 - SMI Referenced Properties/Methods for CIM_SASPHY

Properties	Flags	Requirement	Description & Notes
HardwareMinimumPhysicalLinkRate		Mandatory	
HardwareMaximumPhysicalLinkRate		Mandatory	
ProgrammedMinimumPhysicalLinkRate		Mandatory	
ProgrammedMaximumPhysicalLinkRate		Mandatory	
NegotiatedPhysicalLinkRate		Mandatory	

17.5.20CIM_SASPhyStatistics

Statistics for a SAS PHY.

Requirement: Optional

Table 180 describes class CIM_SASPhyStatistics.

Table 180 - SMI Referenced Properties/Methods for CIM_SASPhyStatistics

Properties	Flags	Requirement	Description & Notes
InvalidDwordCount		Mandatory	
RunningDisparityErrorCount		Mandatory	
LossOfDwordSyncCount		Mandatory	
ResetProblemCount		Mandatory	

IMPLEMENTED

SAS Initiator Ports Profile

DEPRECATED

18 ATA Initiator Ports Profile

The functionality of the ATA Initiator Ports Profile has been removed from this version of the standard.

The Experimental ATA Initiator Ports Profile is defined in Clause 19 of SMI-S 1.6.1

DEPRECATED

ATA Initiator Ports Profile

EXPERIMENTAL

19 FC-SB-x Initiator Ports Profile

19.1 Synopsis

Profile Name: SB Initiator Ports (Component Profile)

Version: 1.7.0

Organization: SNIA

Central Class: FCPort

Scoping Class: a ComputerSystem in a referencing autonomous profile

Related Profiles: Table 181 describes the related profiles for SB Initiator Ports.

Table 181 - Related Profiles for SB Initiator Ports

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

The FC-SB-x Initiator Ports Profile models initiator ports that support the FC-SB-x protocol.

19.2 Description

The FC-SB-x Initiator Ports Profile models initiator ports that support the FC-SB-x protocol.

19.3 Implementation

Figure 27 is an example of a single initiator port. The instance diagram shows a disk (LogicalDevice in the diagram would be subclassed as something like StorageExtent) in an array, connected by a Fibre Channel port. The full model for the disk is shown in the Disk Drive Lite Profile. SBProtocolController is not generally used in initiator contexts. It is included here to be compatible with SMI-S 1.0 clients.

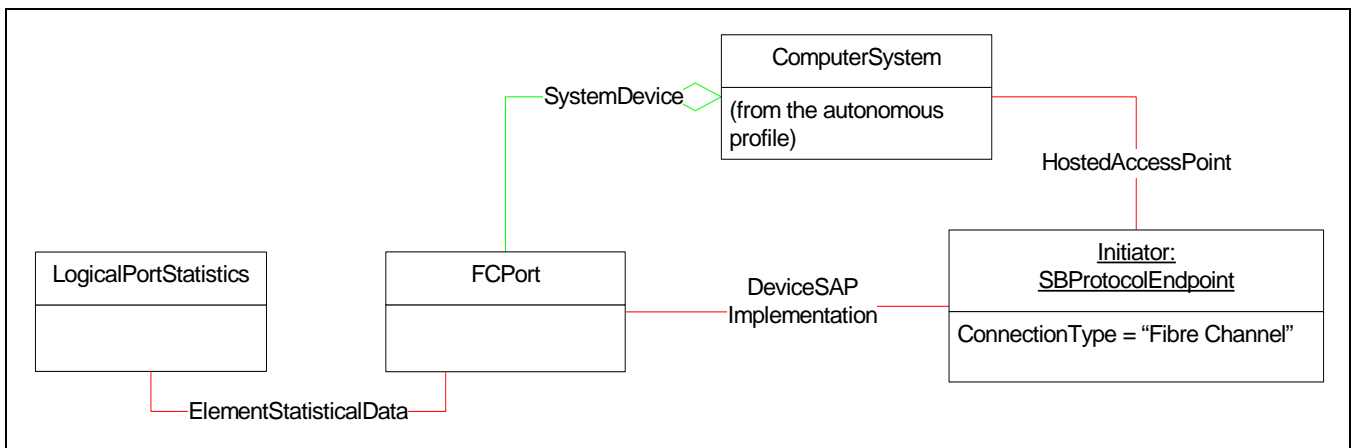


Figure 27 - Fibre Channel Initiator Instance Diagram

19.3.1 Health and Fault Management Considerations

Table 182 summarizes the Health and Fault Management considerations specific to this profile.

Table 182 - FCPort OperationalStatus

OperationalStatus	Description
OK	Port is online
Error	Port has a failure
Stopped	Port is disabled
InService	Port is in Self Test
Unknown	

19.3.2 Cascading Considerations

Not defined in this standard.

19.4 Methods

19.4.1 Extrinsic Methods of the Profile

Not defined in this standard.

19.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

19.5 Client Considerations and Recipes

No recipes are defined in this version of the standard.

19.6 CIM Elements

Table 183 describes the CIM elements for SB Initiator Ports.

Table 183 - CIM Elements for SB Initiator Ports

Element Name	Requirement	Description
19.6.1 CIM_ConnectivityCollection	Optional	Represents a collection of connected SBProtocolEndpoints.
19.6.2 CIM_DeviceSAPImplementation	Mandatory	Connects Initiator SBLogicalPort and SBProtocolEndpoint.
19.6.3 CIM_ElementStatisticalData (Port Statistics)	Optional	Connects SBPort and LogicalPortStatistics.
19.6.4 CIM_FCPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
19.6.5 CIM_HostedAccessPoint (Initiator)	Mandatory	Associates system to initiator protocol endpoints.
19.6.6 CIM_HostedAccessPoint (Target)	Optional	Associates system to optional remote protocol endpoints.
19.6.7 CIM_HostedCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates the ConnectivityCollection to the hosting System.
19.6.8 CIM_MemberOfCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Represents a collection of connected SBProtocolEndpoints.
19.6.9 CIM_SystemDevice (Initiator Ports)	Mandatory	Associates system to initiator ports.
19.6.10 CIM_LogicalPortStatistics	Optional	Statistics for a port.
19.6.11 CIM_SBInitiatorTargetLogicalUnitPath	Optional	
19.6.12 CIM_SBProtocolEndpoint (Initiator)	Mandatory	Represents a protocol (command set) supported by the port. The appropriate subclass (SCSIProtocolEndpoint, ATAPProtocolEndpoint, SBProtocolEndpoint) should be used in initiator port specialized profiles.
19.6.13 CIM_SBProtocolEndpoint (Target)	Optional	Target or non-local ProtocolEndpoint.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_FCPort	Optional	Create FCPort.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::OperationalStatus <> PreviousInstance.CIM_FCPort::OperationalStatus	Optional	CQL -Modify FCPort.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FCPort	Optional	Delete FCPort.

19.6.1 CIM_ConnectivityCollection

Represents a collection of connected SBProtocolEndpoints. The class definition specializes the CIM_ConnectivityCollection definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 184 describes class CIM_ConnectivityCollection.

Table 184 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	

19.6.2 CIM_DeviceSAPImplementation

Connects Initiator SBLogicalPort and SBProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 185 describes class CIM_DeviceSAPImplementation.

Table 185 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SBProtocolEndpoint(Initiator).
Antecedent		Mandatory	Reference to FCPort.

19.6.3 CIM_ElementStatisticalData (Port Statistics)

Connects SBPort and LogicalPortStatistics. The class definition specializes the CIM_ElementStatisticalData definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 186 describes class CIM_ElementStatisticalData (Port Statistics).

Table 186 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to FCPort.
Stats		Mandatory	Reference to LogicalPortStatistics.

19.6.4 CIM_FCPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static

Deleted By: Static
 Requirement: Mandatory

Table 187 describes class CIM_FCPort.

Table 187 - SMI Referenced Properties/Methods for CIM_FCPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 3 for ports restricted to Back-end only or 4 if the port is unrestricted.
PortType		Mandatory	Shall be 0 1 10 11 12 13 14 15 16 17 18 (Unknown or Other or N or NL or F/NL or Nx or E or F or FL or B or G).
ElementName		Mandatory	Port Symbolic Name.
Speed		Mandatory	
MaxSpeed		Mandatory	Port Supported Speed from HBA API.
PortNumber		Optional	
PermanentAddress	CD	Optional	Port WWN. PermanentAddress is optional when used as a backend port in a device. This may be overridden in profiles that use this profile. See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 57.6.3 Standard Formats for Port Names</i> .
NetworkAddresses		Optional	For Fibre Channel end device ports, the Fibre Channel ID.
LinkTechnology		Mandatory	
SupportedMaximumTransmissionUnit		Mandatory	

19.6.5 CIM_HostedAccessPoint (Initiator)

Associates system to initiator protocol endpoints.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 188 describes class CIM_HostedAccessPoint (Initiator).

Table 188 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to ProtocolEndpoint(Initiator).

19.6.6 CIM_HostedAccessPoint (Target)

Associates system to optional remote protocol endpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 189 describes class CIM_HostedAccessPoint (Target).

Table 189 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to ProtocolEndpoint(Target).

19.6.7 CIM_HostedCollection (Connectivity Collection)

Associates the ConnectivityCollection to the hosting System.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 190 describes class CIM_HostedCollection (Connectivity Collection).

Table 190 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ConnectivityCollection.
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.

19.6.8 CIM_MemberOfCollection (Connectivity Collection)

Represents a collection of connected SBProtocolEndpoints. The class definition specializes the CIM_MemberOfCollection definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 191 describes class CIM_MemberOfCollection (Connectivity Collection).

Table 191 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Member		Mandatory	Reference to SBProtocolEndpoint(Initiator or Target).
Collection		Mandatory	Reference to ConnectivityCollection.

19.6.9 CIM_SystemDevice (Initiator Ports)

Associates system to initiator ports.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 192 describes class CIM_SystemDevice (Initiator Ports).

Table 192 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to LogicalPort.

19.6.10 CIM_LogicalPortStatistics

Statistics for a port.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 193 describes class CIM_LogicalPortStatistics.

Table 193 - SMI Referenced Properties/Methods for CIM_LogicalPortStatistics

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
InstanceID		Mandatory	
BytesTransmitted		Mandatory	
BytesReceived		Mandatory	
PacketsTransmitted		Mandatory	
PacketsReceived		Mandatory	

19.6.11 CIM_SBInitiatorTargetLogicalUnitPath

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 194 describes class CIM_SBInitiatorTargetLogicalUnitPath.

Table 194 - SMI Referenced Properties/Methods for CIM_SBInitiatorTargetLogicalUnitPath

Properties	Flags	Requirement	Description & Notes
UsePreferredPath		Optional	Boolean indicating whether preferred path processing is required.
PreferredPath		Optional	Boolean indicating whether this is a preferred path.
PathGroupState		Optional	One of 0(Unknown), 2(Path grouping not supported), 3(Reset), 4(Grouped), or 5(Ungrouped).
PathGroupMode		Optional	One of 0(Unknown), 2(None), 3(Single path), or 4(Multipath). Single path and multipath only valid if PathGroupState is grouped.
PathGroupID		Optional	String containing the ID from the OS, only valid if PathGroupState is Grouped.
LogicalUnit		Mandatory	Reference to StorageExtent in Disk Drive Lite Profile or MediaAccessDevice in Media Access Device Profile.
Target		Mandatory	Reference to SCSIProtocolEndpoint(Target).
Initiator		Mandatory	Reference to SCSIProtocolEndpoint(Initiator).

19.6.12 CIM_SBProtocolEndpoint (Initiator)

Represents a protocol (command set) supported by the port. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 195 describes class CIM_SBProtocolEndpoint (Initiator).

Table 195 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	C	Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 57.6.3 Standard Formats for Port Names.</i>
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be 'SB'.

Table 195 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).
Role		Mandatory	Shall be 2 (Initiator) or 4 (Both Initiator and Target).

19.6.13 CIM_SBProtocolEndpoint (Target)

Target or non-local ProtocolEndpoint. The class definition specializes the CIM_ProtocolEndpoint definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 196 describes class CIM_SBProtocolEndpoint (Target).

Table 196 - SMI Referenced Properties/Methods for CIM_SBProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be 'SB'.
Role		Mandatory	Should be set appropriately by the instrumentation. If not know, use 0 (Unknown).
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).

EXPERIMENTAL

EXPERIMENTAL

20 FCoE Initiator Ports Profile

20.1 Synopsis

Profile Name: FCoE Initiator Ports (Component Profile)

Version: 1.6.0

Organization: SNIA

Specializes: Generic Initiator Ports Profile

Central Class: FCPort

Scoping Class: ComputerSystem in the Base Server Profile (or some other autonomous profile)

Related Profiles: Not defined in this standard.

The FCoE Initiator Ports Profile models the behavior of the Fibre Channel over Ethernet (FCoE) functionality of a Converged Network Adaptor (CNA).

20.2 Description

The FCoE Initiator Ports Profile is a component profile that models the behavior of the Fibre Channel over Ethernet (FCoE) functionality of a Converged Network Adaptor (CNA).

A CNA may support functionality beyond FCoE, including generic TCP/IP support. Functionality of CNAs other than FCoE is outside the scope of this profile.

20.3 Implementation

Figure 28: "FCoE Initiator Instance Diagram" shows the model for classes in the FCoE Initiator Ports Profile

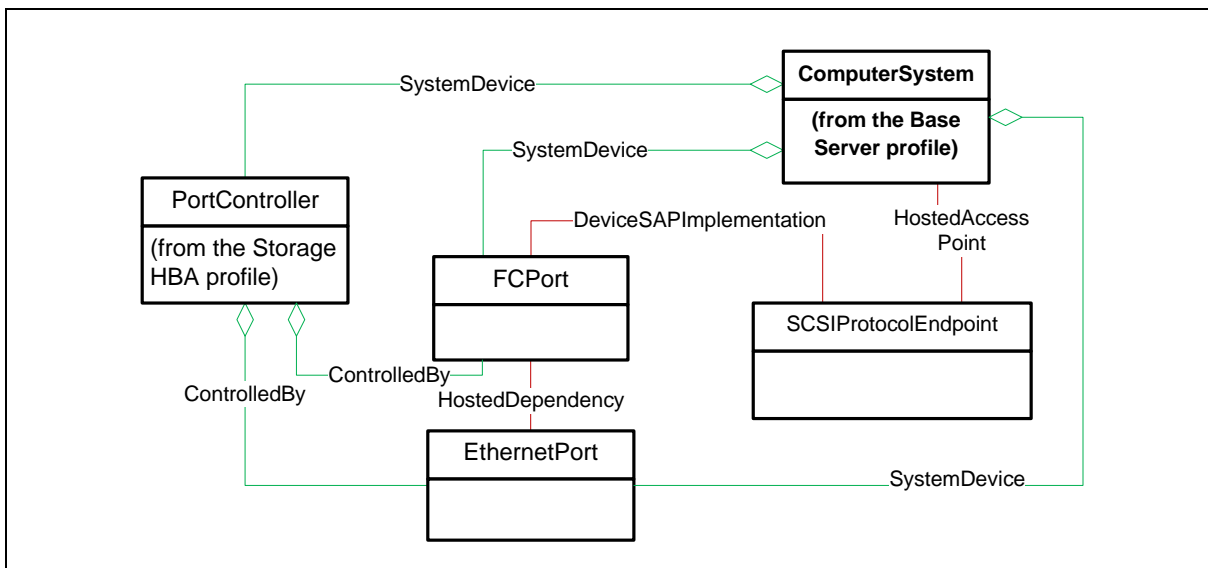


Figure 28 - FCoE Initiator Instance Diagram

Each FCoE port shall be modeled with single instance each of FCPort, EthernetPort, and SCSIProtocolEndpoint. FCPort and EthernetPort shall be associated with HostedDependency. FCPort

and EthernetPort shall each be associated to the ComputerSystem in the Base Server Profile with SystemDevice. FCPort and SCSIProtocolEndpoint shall be associated with DeviceSAPImplementation. SCSIProtocolEndpoint shall be associated to the ComputerSystem (defined in the Base Server Profile) with HostedAccessPoint.

20.3.1 Relationship to Storage HBA Profile

The FCoE Initiator ports profile is used in conjunction with the Storage HBA Profile (see *Storage Management Technical Specification, Part 7 Host Elements, 1.7.0 Rev 5 6 Storage HBA Profile*). The Storage HBA Profile models the management of HBA cards independent of connectivity, and the FCoE Ethernet Ports Profile models the management of FCoE ports on an HBA.

The ControlledBy association defined in *Storage Management Technical Specification, Part 7 Host Elements, 1.7.0 Rev 5 6 Storage HBA Profile* shall reference instance of FCPort and shall not reference instances of EthernetPort.

20.3.2 Optional target model

Figure 29: "Optional Target Element Model" shows an example of a single port and drive connected to a single system using Fibre Channel. This instance diagram shows a disk (LogicalDevice in the diagram would be subclassed as something like StorageExtent) in an array, connected by a Fibre Channel port. The full model for the disk is shown in 10 Disk Drive Lite Profile.

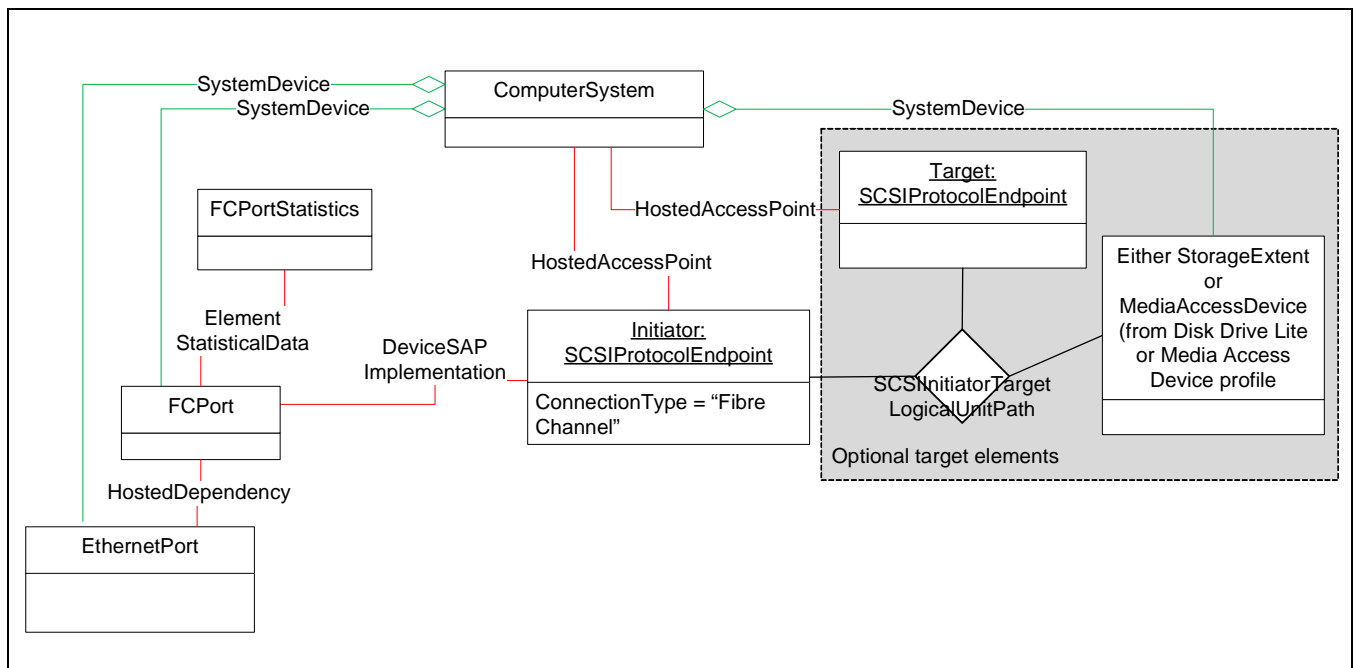


Figure 29 - Optional Target Element Model

20.3.3 Port Statistics

The FCPortStatistics subclass of NetworkPortStatistics is optional. If supported, FCPortStatistics shall be associated to FcPort using ElementStatisticalData.

20.3.4 Logical Port Group (FC Node)

LogicalPortGroup may optionally be used to model the collection of ports that shared a Node WWN (in this case, both ports on a card, but other implementations are in use). If LogicalPortGroup is instantiated,

it shall be associated to the ComputerSystem in the referencing profile using HostedCollection and also associated to FCPorts using MemberOfCollection.

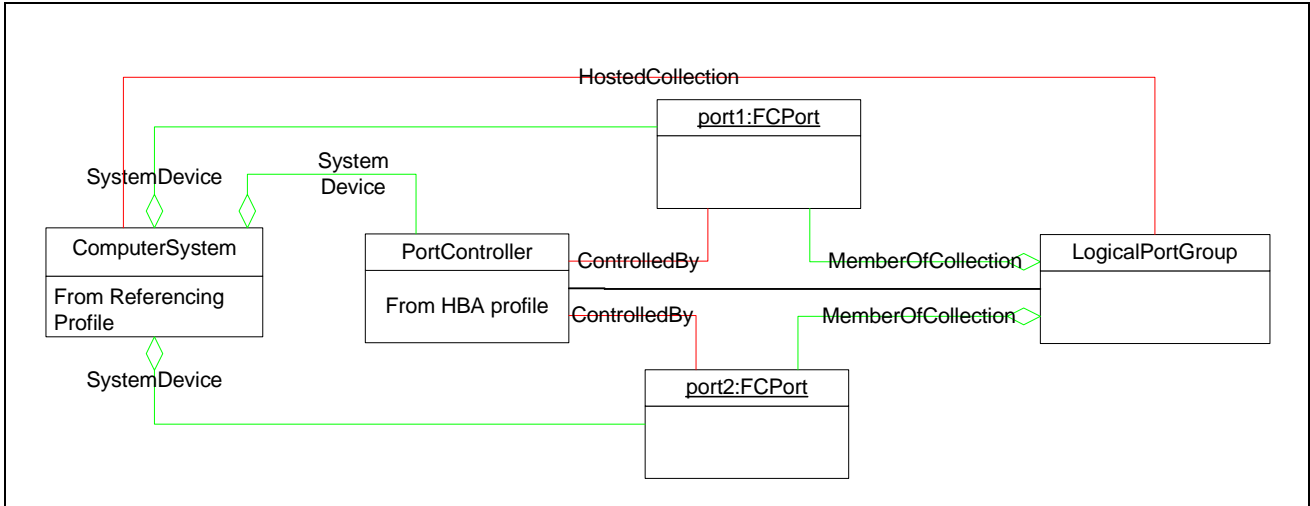


Figure 30 - Logical Port Group Model

20.3.5 Health and Fault Management Considerations

Table 197 summarized the Health and Fault Management considerations specific to this profile.

Table 197 - FCPort OperationalStatus

OperationalStatus	Description
(2) OK	Port is online
(6) Error	Port has a failure
(10) Stopped	Port is disabled
(11) InService	Port is in Self Test
(0) Unknown	

20.3.6 Cascading Considerations

Not defined in this standard.

20.4 Methods

20.4.1 Extrinsic Methods of this Profile

Not defined in this standard

20.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported

are as follows:

- GetInstance
- Associators

- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

20.5 Use Cases and Recipes

No recipes are defined in this version of the standard.

20.6 CIM Elements

Table 198 describes the CIM elements for FCoE Initiator Ports.

Table 198 - CIM Elements for FCoE Initiator Ports

Element Name	Requirement	Description
20.6.1 CIM_ConnectivityCollection	Optional	Represents a collection of connected ProtocolEndpoints.
20.6.2 CIM_DeviceSAPImplementation	Mandatory	Connects Initiator LogicalPort and ProtocolEndpoint.
20.6.3 CIM_ElementStatisticalData (Port Statistics)	Optional	Connects LogicalPort and LogicalPortStatistics.
20.6.4 CIM_EthernetPort	Mandatory	
20.6.5 CIM_FCPort	Mandatory	Represents the logical aspects of the physical port and may have multiple associated protocols.
20.6.6 CIM_FCPortStatistics	Mandatory	Statistics for a port.
20.6.7 CIM_HostedAccessPoint (Initiator)	Mandatory	Associates system to initiator protocol endpoints.
20.6.8 CIM_HostedAccessPoint (Target)	Optional	Associates system to optional remote protocol endpoints.
20.6.9 CIM_HostedCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates the ConnectivityCollection to the hosting System.
20.6.10 CIM_HostedCollection (FC Node)	Optional	Associates the LogicalPortGroup (Fibre Channel Node) to the hosting System.
20.6.11 CIM_HostedDependency (NetworkPort to FCPort)	Mandatory	Association between EthernetPort and FCPort.
20.6.12 CIM_LogicalPortGroup	Optional	Collection of Fibre Channel ports that share a Node WWN.
20.6.13 CIM_MemberOfCollection (Connectivity Collection)	Conditional	Conditional requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections. Associates ProtocolEndpoints to the ConnectivityCollection.
20.6.14 CIM_MemberOfCollection (FC Node)	Optional	Associates FCPort to the LogicalPortGroup.
20.6.15 CIM_ProtocolEndpoint (Initiator)	Mandatory	Represents a protocol (command set) supported by the port. The appropriate subclass (SCSIProtocolEndpoint, ATAProtocolEndpoint, SBProtocolEndpoint) should be used in initiator port specialized profiles.
20.6.16 CIM_ProtocolEndpoint (Target)	Optional	Models protocols of remote ports - target devices and possibly other initiators.

Table 198 - CIM Elements for FCoE Initiator Ports

Element Name	Requirement	Description
20.6.17 CIM_SCSIInitiatorTargetLogicalUnitPath	Optional	Represents a path between a SCSI initiator, target, and logical unit.
20.6.18 CIM_SCSIProtocolEndpoint (Initiator)	Mandatory	
20.6.19 CIM_SCSIProtocolEndpoint (Target)	Optional	Models remote ports - target devices and possibly other initiators.
20.6.20 CIM_SystemDevice (Ethernet Port)	Mandatory	Associates system to ethernet ports.
20.6.21 CIM_SystemDevice (Initiator Ports)	Mandatory	Associates system to FCPorts.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_FCPort	Optional	CQL -Creation of an FCoE Port.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_FCPort AND SourceInstance.CIM_FCPort::OperationalStatus <> PreviousInstance.CIM_FCPort::OperationalStatus	Optional	CQL -Modify FCPort.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FCPort	Optional	CQL -Deletion of an FCoE Port.

20.6.1 CIM_ConnectivityCollection

Represents a collection of connected ProtocolEndpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 199 describes class CIM_ConnectivityCollection.

Table 199 - SMI Referenced Properties/Methods for CIM_ConnectivityCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	

20.6.2 CIM_DeviceSAPImplementation

Connects Initiator LogicalPort and ProtocolEndpoint. The class definition specializes the CIM_DeviceSAPImplementation definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 200 describes class CIM_DeviceSAPImplementation.

Table 200 - SMI Referenced Properties/Methods for CIM_DeviceSAPImplementation

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to SCSIProtocolEndpoint.
Antecedent		Mandatory	Reference to FCPort.

20.6.3 CIM_ElementStatisticalData (Port Statistics)

Connects LogicalPort and LogicalPortStatistics. The class definition specializes the CIM_ElementStatisticalData definition in the Generic Initiator Ports profile.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 201 describes class CIM_ElementStatisticalData (Port Statistics).

Table 201 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Port Statistics)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to FC subclass.
Stats		Mandatory	Reference to FCPortStatistics.

20.6.4 CIM_EthernetPort

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 202 describes class CIM_EthernetPort.

Table 202 - SMI Referenced Properties/Methods for CIM_EthernetPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
LinkTechnology		Mandatory	Shall be 2 (Ethernet).
OperationalStatus		Mandatory	Shall be 0 (Unknown), 2 (OK), 6 (Error), 10 (Stopped), or 11 (In Service).
PermanentAddress	CD	Mandatory	The MAC Address. Shall be formatted as 12 un-separated upper case hex digits.

20.6.5 CIM_FCPort

Represents the logical aspects of the physical port and may have multiple associated protocols. The class definition specializes the CIM_LogicalPort definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 203 describes class CIM_FCPort.

Table 203 - SMI Referenced Properties/Methods for CIM_FCPort

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
OperationalStatus		Mandatory	
UsageRestriction		Mandatory	Shall be 3 for ports restricted to back-end (initiator) only or 4 if the port is unrestricted.
PortType		Mandatory	Shall be 0 (Unknown), 1 (Other), N (10), or Nx (13).
OtherPortType		Conditional	Required when PortType is Other
ElementName		Mandatory	Port Symbolic Name.
Speed		Mandatory	Speed in bits per second. Shall be 0, 1062500000 (1GFC), 2125000000 (2GFC), 4250000000 (4GFC), 8500000000 (8GFC), 10312500000 (10GE/10GFCoE), 14025000000 (16GFC), or 28500000000 (32GFC).
MaxSpeed		Mandatory	Maximum Port Speed.
PermanentAddress	CD	Optional	Port WWN. PermanentAddress is optional when used as a back-end port in a device. This may be overridden in profiles that use this profile. Shall be 16 un-separated upper case hex digits.
NetworkAddresses		Optional	For Fibre Channel end device ports, the Fibre Channel ID. Shall be 16 un-separated upper case hex digits.
LinkTechnology		Mandatory	Shall be 4 (FC).
SupportedMaximumTransmissionUnit		Mandatory	
PortDiscriminator		Mandatory	Experimental. Shall include 10 (FCoE) and 12 (HBA).

20.6.6 CIM_FCPortStatistics

The class definition specializes the CIM_LogicalPortStatistics definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 204 describes class CIM_FCPortStatistics.

Table 204 - SMI Referenced Properties/Methods for CIM_FCPortStatistics

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
InstanceID		Mandatory	
BytesTransmitted		Mandatory	
BytesReceived		Mandatory	
PacketsTransmitted		Mandatory	
PacketsReceived		Mandatory	
CRCErrors		Mandatory	Maps to HBA API HBA_PortStatistics.InvalidCRCCCount.
LinkFailures		Mandatory	Maps to HBA API HBA_PortStatistics.LinkFailureCount.
PrimitiveSeqProtocolErrCo unt		Mandatory	
LossOfSignalCounter		Mandatory	Maps to HBA API HBA_PortStatistics.LossOfSignalCount.
InvalidTransmissionWords		Mandatory	Maps to HBA API HBA_PortStatistics.InvalidTxWordCount.
StatisticTime		Optional	Time last measurement was taken.
LIPCount		Mandatory	
NOSCount		Mandatory	
ErrorFrames		Mandatory	
DumpedFrames		Mandatory	
LossOfSyncCounter		Mandatory	Maps to HBA API HBA_PortStatistics.LossOfSynchCount.

20.6.7 CIM_HostedAccessPoint (Initiator)

Associates system to initiator protocol endpoints.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 205 describes class CIM_HostedAccessPoint (Initiator).

Table 205 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Initiator)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to ProtocolEndpoint(Initiator).

20.6.8 CIM_HostedAccessPoint (Target)

Associates system to optional remote protocol endpoints.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 206 describes class CIM_HostedAccessPoint (Target).

Table 206 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint (Target)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.
Dependent		Mandatory	Reference to ProtocolEndpoint(Target).

20.6.9 CIM_HostedCollection (Connectivity Collection)

Associates the ConnectivityCollection to the hosting System.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 207 describes class CIM_HostedCollection (Connectivity Collection).

Table 207 - SMI Referenced Properties/Methods for CIM_HostedCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to ConnectivityCollection.
Antecedent		Mandatory	Reference to ComputerSystem in referencing profile.

20.6.10 CIM_HostedCollection (FC Node)

Associates the LogicalPortGroup (Fibre Channel Node) to the hosting System.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 208 describes class CIM_HostedCollection (FC Node).

Table 208 - SMI Referenced Properties/Methods for CIM_HostedCollection (FC Node)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem.
Dependent		Mandatory	Reference to LogicalPortGroup.

20.6.11 CIM_HostedDependency (NetworkPort to FCPort)

Association between EthernetPort and FCPort.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 209 describes class CIM_HostedDependency (NetworkPort to FCPort).

Table 209 - SMI Referenced Properties/Methods for CIM_HostedDependency (NetworkPort to FCPort)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to EthernetPort.
Dependent		Mandatory	Reference to FCPort.

20.6.12 CIM_LogicalPortGroup

Represents the Fibre Channel Node.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 210 describes class CIM_LogicalPortGroup.

Table 210 - SMI Referenced Properties/Methods for CIM_LogicalPortGroup

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	Opaque.
Name	D	Mandatory	Fibre Channel Node WWN.
NameFormat		Mandatory	Shall be 'WWN'.
ElementName		Mandatory	Node Symbolic Name.

20.6.13 CIM_MemberOfCollection (Connectivity Collection)

Associates ProtocolEndpoints to the ConnectivityCollection. The class definition specializes the CIM_MemberOfCollection definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ConnectivityCollections that are not RemoteReplicationCollections.

Table 211 describes class CIM_MemberOfCollection (Connectivity Collection).

Table 211 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Connectivity Collection)

Properties	Flags	Requirement	Description & Notes
Member		Mandatory	Reference to ProtocolEndpoint.
Collection		Mandatory	Reference to ConnectivityCollection.

20.6.14 CIM_MemberOfCollection (FC Node)

Associates FCPort to the LogicalPortGroup.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 212 describes class CIM_MemberOfCollection (FC Node).

Table 212 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (FC Node)

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	Reference to LogicalPortGroup.
Member		Mandatory	Reference to FCPort.

20.6.15 CIM_ProtocolEndpoint (Initiator)

Represents a protocol (command set) supported by the port.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 213 describes class CIM_ProtocolEndpoint (Initiator).

Table 213 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name	C	Mandatory	See <i>Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5</i> 7.6.3 Standard Formats for Port Names.
ProtocollFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI', 'ATA', or 'SB'. Initiator port specialized profiles specify the appropriate subset.

20.6.16 CIM_ProtocolEndpoint (Target)

Models protocols of remote ports - target devices and possibly other initiators. The appropriate subclass (SCSIProtocolEndpoint, ATAProtocolEndpoint, SBProtocolEndpoint) should be used in initiator port specialized profiles.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 214 describes class CIM_ProtocolEndpoint (Target).

Table 214 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	The values in MOFs map to IETF values and exclude storage. Shall be 1 (Other) and set OtherTypeDescription appropriately.
OtherTypeDescription		Mandatory	Shall be the string 'SCSI', 'ATA', or 'SB'. Initiator port specialized profiles specify the appropriate subset.

20.6.17 CIM_SCSIInitiatorTargetLogicalUnitPath

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 215 describes class CIM_SCSIInitiatorTargetLogicalUnitPath.

Table 215 - SMI Referenced Properties/Methods for CIM_SCSIInitiatorTargetLogicalUnitPath

Properties	Flags	Requirement	Description & Notes
LogicalUnit		Mandatory	
Initiator		Mandatory	
Target		Mandatory	

20.6.18 CIM_SCSIProtocolEndpoint (Initiator)

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 216 describes class CIM_SCSIProtocolEndpoint (Initiator).

Table 216 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Initiator)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ProtocolIFType		Mandatory	Shall be 1 (Other).
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 2 (Fibre Channel).
Role		Mandatory	Shall be 2 (Initiator).

20.6.19 CIM_SCSIProtocolEndpoint (Target)

Models remote ports - target devices and possibly other initiators.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 217 describes class CIM_SCSIProtocolEndpoint (Target).

Table 217 - SMI Referenced Properties/Methods for CIM_SCSIProtocolEndpoint (Target)

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
Role		Mandatory	Should be set appropriately by the instrumentation. If not know, use 0 (Unknown).
ProtocolIFType		Mandatory	The values in MOFs map to IETF values and exclude storage. Shall be 1 (Other) and set OtherTypeDescription to 'SCSI'.
OtherTypeDescription		Mandatory	Shall be the string 'SCSI'.
ConnectionType		Mandatory	Shall be 8 (FC).

20.6.20 CIM_SystemDevice (Ethernet Port)

Associates system to ethernet ports.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 218 describes class CIM_SystemDevice (Ethernet Port).

Table 218 - SMI Referenced Properties/Methods for CIM_SystemDevice (Ethernet Port)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to EthernetPort.

20.6.21 CIM_SystemDevice (Initiator Ports)

Associates system to FCPorts. The class definition specializes the CIM_SystemDevice definition in the Generic Initiator Ports profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 219 describes class CIM_SystemDevice (Initiator Ports).

Table 219 - SMI Referenced Properties/Methods for CIM_SystemDevice (Initiator Ports)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem.
PartComponent		Mandatory	Reference to FCPort.

EXPERIMENTAL

STABLE

21 Access Points Profile

21.1 Synopsis

Profile Name: Access Points (Component Profile)

Version: 1.3.0

Organization: SNIA

Central Class: RemoteServiceAccessPoint

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

21.2 Description

The Access Points Profile provides addresses of remote access points for management services.

This is modeled using a RemoteServiceAccessPoint linked to the managed system using a HostedAccessPoint association.

A management service is typically associated with all elements in a system, but in some cases, a management service relates to a subset of elements. The scope of a RemoteServiceAccessPoint may be constrained to a subset of elements using SAPAvailableForElement. If the service referenced in RemoteServiceAccessPoint is not referenced by any SAPAvailableForElement associations, then the service described by RemoteServiceAccessPoint shall apply to all the elements of the system referenced via HostedAccessPoints. This type of system-wide service is depicted in Figure 31: "System-wide Remote Access Point".

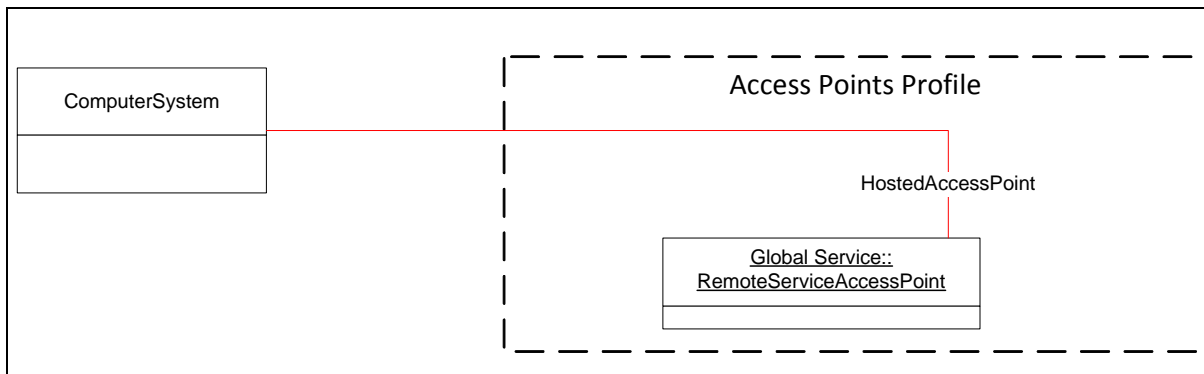


Figure 31 - System-wide Remote Access Point

If the service referenced in RemoteServiceAccessPoint is referenced by any SAPAvailableForElement associations, then the service described by RemoteServiceAccessPoint shall apply to the subset of elements referenced via SAPAvailableForElement associations. The HostedAccessPoint association between RemoteServiceAccessPoint is still mandatory (so the client can readily associate the service to a specific storage system).

Figure 32: "Access Point Instance Diagram" depicts a configuration with two RemoteServiceAccessPoint instances. One represents a system-wide service and the other represents a service that applies just to certain devices.

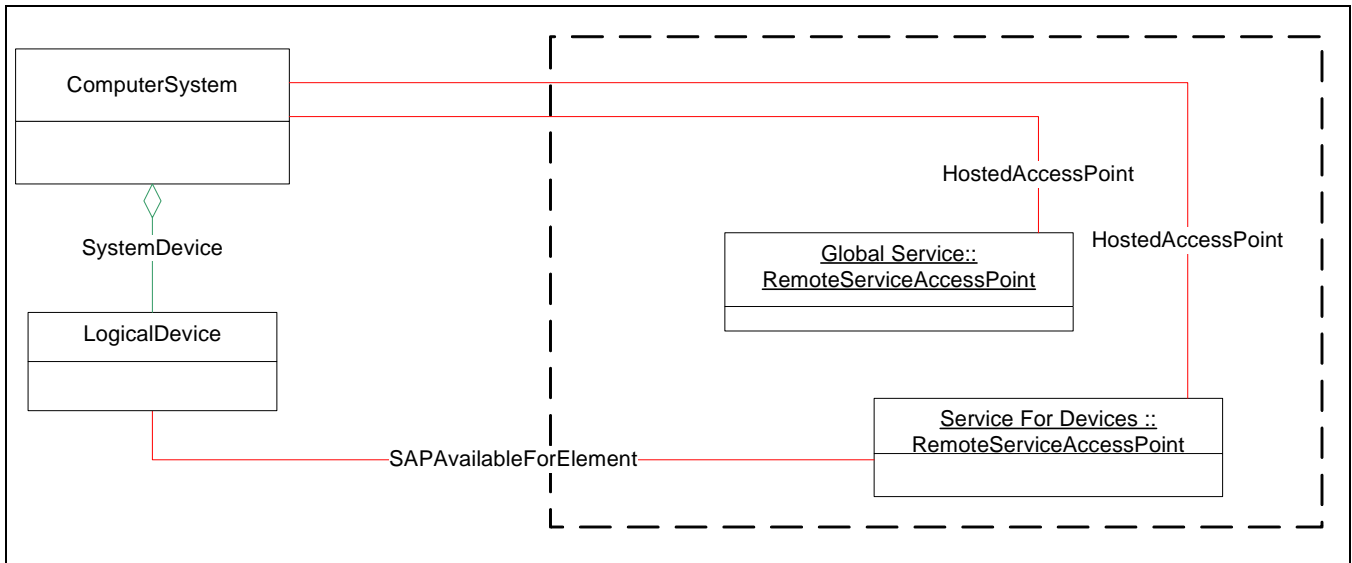


Figure 32 - Access Point Instance Diagram

The exposed management services may represent a web UI that can be launched by a web browser, a telnet interface, or some vendor-specific interface. RemoteServiceAccessPoint InfoFormat property describes the format of the AccessInfo property; valid options include "URL" and FQDN". In a URL, the text before the "://" is referred to as the "scheme". A URL with an http or HTTPS scheme is often a web/HTML page, but HTTP can be used for other purposes. Table 220 specifies the requirements for InfoFormat, AccessInfo, and the scheme subset of a URL AccessInfo.

Table 220 - RemoteAccessPoint InfoFormat and AccessInfo Properties

InfoFormat	AccessInfo Scheme	Description
"URL"	"http" or "https"	The references URL shall be a valid web page. It should provide element management for the system or elements referenced by the associated HostedAccessPoint association.
"Other" with OtherInfoFormatDescription = "Non-UI URL"	"http" or "https"	Used for HTTP URLs that do not reference a valid web UI.
"URL"	anything other than "http" and "https"	May be used. No standard behavior is specified.
others from the MOF	n/a	May be used. No standard behavior is specified.

21.3 Health and Fault Management Considerations

Not defined in this standard.

21.4 Cascading Considerations

Not defined in this standard.

21.5 Methods of this Profile

Not defined in this standard.

21.6 Client Considerations and Recipes

Not defined in this standard.

21.7 CIM Elements

Table 221 describes the CIM elements for Access Points.

Table 221 - CIM Elements for Access Points

Element Name	Requirement	Description
21.7.1 CIM_HostedAccessPoint	Mandatory	Associate the RemoteServiceAccessPoint to the System on which it is hosted.
21.7.2 CIM_RemoteServiceAccessPoint	Mandatory	A ServiceAccessPoint for management tools.
21.7.3 CIM_SAPAvailableForElement	Optional	This association identifies the element that is serviced by the RemoteServiceAccessPoint.

21.7.1 CIM_HostedAccessPoint

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 222 describes class CIM_HostedAccessPoint.

Table 222 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	The Hosting System.
Dependent		Mandatory	The access point(s) that are hosted on this System.

21.7.2 CIM_RemoteServiceAccessPoint

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 223 describes class CIM_RemoteServiceAccessPoint.

Table 223 - SMI Referenced Properties/Methods for CIM_RemoteServiceAccessPoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
CreationClassName		Mandatory	
SystemName		Mandatory	
Name		Mandatory	
ElementName		Mandatory	User Friendly name.

Table 223 - SMI Referenced Properties/Methods for CIM_RemoteServiceAccessPoint

Properties	Flags	Requirement	Description & Notes
AccessInfo		Mandatory	Management Address.
InfoFormat		Mandatory	The format of the Management Address. For interoperability, this shall be 'URL' (200).

21.7.3 CIM_SAPAvailableForElement

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 224 describes class CIM_SAPAvailableForElement.

Table 224 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	The managed element.
AvailableSAP		Mandatory	The service access point.

STABLE

STABLE**22 Health Package****22.1 Synopsis**

Profile Name: Health Package (Component Profile)

Version: 1.2.0

Organization: SNIA

Central Class: N/A

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

22.2 Description

Failures and abnormal occurrences are a common and expected part of monitoring, controlling, and configuring devices and applications. A SMI-S client needs to be prepared at all times to trap unexpected situations and take appropriate action. This package defines the general mechanisms used in the expression of health in SMI-S. This package does not define the particular way a particular profile or package reports health.

This package builds on *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5* Health and Fault Management. In particular, this package defines the basis of all the sections that currently and will exist in this specification or future versions of same.

22.2.1 Error Reporting Mechanism

Error are reports for many reasons. Not all the reasons are directly related to the operation being imposed on the implementation by the client. It is therefore necessary for the client to be able to distinguish between errors that are associated to problems in the formation and invocation of a method, extrinsic or intrinsic, or are related to other conditions.

The client application may need to reform the method call itself, by fixing parameters for example, or the client may need to stop what its attempting. At a basic level, the client needs to know that this operation will succeed at all, given the prevailing conditions on the managed element. A client may also need to notify the end-user of the situation that is preventing the client from fulfilling its function. A HFM application may need to investigate the failure and develop a prognosis.

The types of errors are categorized in the three following types.

- a) Errors associated to the method call
- b) Errors caused by adverse prevailing conditions in the managed element
- c) Errors causes by adverse prevailing conditions in the WBEM Server or related, infrastructural components

Obviously, the method called may not exist. There may be a spelling mistake for the method name. One or more of the parameters may be incorrectly formed, expressed, or otherwise invalid. The first type of error, type a, is designed to inform the client that the operation attempted is still valid, but that the request was faulty. The intent of such an error is to tell the client what is wrong with the method call and allow the method to be invoked again.

On the other hand, the device or application may be in some failure condition which prevents it from honoring this particular or several method calls. This type of error, type b, tells the client that the it is unlikely that the method being attempted will be honored. Specifically, the method execution is blocked by the prevailing condition being described in the error itself. Given the presence of both type a and type b error situations, the implementations should report the type b error. In this case, it does not matter how many fixes are made to the method call, the method call will fail anyway.

The WBEM Service is a separate architectural element from the managed element itself. It can fail, even though the methods and the managed element itself are without error. For example, the WBEM Server may allow only a limited number of concurrent connection or request and reject all others. The server may be shutting down or starting up and thus be unable to process any requests at the time. Unlike type b errors, type c errors are usually transient in nature. Since a failure in the WBEM Server or its components constitutes a communications failure, the reporting of type c errors shall take precedence over all other existing error type conditions.

The WBEM Server returns a error response or a results response to the request, which contains the operation previously mentioned. Errors in WBEM may be reported through two ways. The status code itself provides basic failure information. The number of status codes is very limited. Also on conveyed on the error response, is a Error instance. The Error provides vastly most information than the status code and, as such, is a superior mechanism for reporting errors.

The CIM Error provides attributes to express the categorization and severity of the error. More importantly, the CIM Error and AlertIndication, to be discussed later, contain the exact expression of the nature of the error and additional parameters to that error.

22.2.2 Event Reporting Mechanism

It is not sufficient to simply report the adverse conditions of the device or application through the error reporting mechanism. Many of the adverse conditions that would be reported to a client application attempting control or configuration operations are also of interest to client applications monitoring the very same device or application.

The CIM Event model provides a special class for reporting event conditions, AlertIndication. The AlertIndication is used to report a device or application conditions that may also be represented in one or more other instances. When the implementation detects the presence of a supported condition, it generates an AlertIndication to those listening clients.

It is recommended that the type b and type c errors reported in also be reported through AlertIndications.

22.2.3 Standard Events

The expression of Error or an Alertindication is not entirely meaningful to the SMI-S client without the standardization. A client can use these classes to determine the category, severity, and some other characteristics of the event, but the client can not determine the exact nature of the event without this standardization.

Standard events are registered and this registry is maintained by some organization or company, like SNIA.

Primary event identification and characterization properties:

- **OwningEntity**
This property defines the registration entity for the event. The entities that are in scope for SMI-S are "DMTF" and "SNIA". If the OwningEntity is neither of these, then this specification provides no meaning for this event.
- **MessageID**
This property defines an event identifier that is unique for the OwningEntity. The combination of the OwningEntity and MessageID defines the entry in the registry.

- **Message**
This property contains the message that can be forwarded to the end-user. The message is built from using the static, `MessageFormatString`, and dynamic, `MessageArguments`, components. This text may be localized. This text is not intended for programmatic processing
- **MessageArguments**
This property defines the variable content for the message. The client would programmatically process the arguments to get further details on the nature of the event. For example, the message argument can tell the client which method parameter has a problem and what the problem is.
- **MessageFormatString**
This property defines the static component of the message. This property is not included in the event instance itself and is only present in the event registry.

22.2.4 Reporting Health

Many devices or applications can attempt to fix themselves upon encountering some adverse condition. The set of components which the device or application can attempt to fix is called the Fault Region. The set may include part or all of other devices or applications. Having the Fault Regions declared helps a HFM application, acting as a doctor, to do no harm by attempting to interfere and thereby adversely effect the corrective action being attempted.

When components fail or become degraded, they can cause other components to fail or become degraded. For an HFM application to report or attempt to diagnose the problem, the device or application should express what the cause and effect relationships are that define the extent of the components affected by the failure or degradation. The `RelatedElementCausingError` class provides just such a mechanism.

The cause and effect relationships identified by the `RelatedElementCausingError` association may be a chain of cause and effect relationships with many levels. Given that devices or applications are sometimes subject to several levels of decomposition, each level of may have its own set of these associations that represent the ranking of cause and effect relationships and their effect on the parent component on the given level.

22.2.5 Computer System Operational Status

For most profiles, the `ComputerSystem` class is used to define the top or head of the object hierarchy. A profile may allow for partitioning or clustering by having more than one `ComputerSystem`, but one `ComputerSystem` often represents the device or application representation. In this role, it is important the summary of the health of the device or application is declared in the `ComputerSystem` instance.

Table 225 - OperationalStatus Details

Primary Operational Status	Subsidiary Operational Status	Description
2 "OK"		The system has a good status.
2 "OK"	4 "Stressed"	The system is stressed, for example the temperature is over limit or there is too much IO in progress.
2 "OK"	5 "Predictive Failure"	The system will probably fail sometime soon.
3 "Degraded"		The system is operational but not at 100% redundancy. A component has suffered a failure or something is running slow.
6 "Error"		An error has occurred causing the system to stop. This error may be recoverable with operator intervention.

Table 225 - OperationalStatus Details (Continued)

Primary Operational Status	Subsidiary Operational Status	Description
6 "Error"	7 "Non-recoverable error"	A severe error has occurred. Operator intervention is unlikely to fix it.
6 "Error"	16 "Supporting entity in error"	A modeled element has failed.
12 "No contact"		The provider knows about the array but has not talked to it since last reboot.
13 "Lost communication"		The provider used to be able to communicate with the array, but has now lost contact.
8 "Starting"		The system is starting up.
9 "Stopping"		The system is shutting down.
10 "Stopped"		The data path is OK but shut down, the management channel is still working.

OperationalStatus is an array. The primary and subsidiary statuses are both OperationalStatus property, and are summarized in Table 225. If the subsidiary operational status is present in the array, it is intended to provide additional clarification to the primary operational status. The implementation shall report one of the above combinations of statuses. It may also report additional statuses beyond the ones defined in Table 225.

The operational status combinations listed in Table 225 that include descriptions about "provider" (i.e., the CIM Provider), are only valid in those cases where the implementation of SMI-S employs a proxy provider.

The operational statuses listed in Table 225 shall not be used to report the status of the WBEM Server itself.

22.2.6 Event Reporting

The implementation may report Event or AlertIndication instances. The profile or package that includes this package defines whether or not these events are supported and when the events are produced.

If the support Event or AlertIndication is implemented, then the implementation shall also support the common messages through both Errors and AlertIndications. This means that the implementation produce the common event listed in the registry when the condition, also described in the registry, is present.

It is mandatory to report error conditions through both AlertIndication or Lifecycle indication and Error in those cases where Error is returned when the method call failed for reasons other than the method call itself. For example, if the device is over heated, then a method call can fail because of this condition. It is expected that the device will report an over heat AlertIndication to listening clients as well.

EXPERIMENTAL

22.2.7 Fault Region

If the device or application is itself attempting to rectify an adverse condition reported through a standard error, then the implementation shall report what corrective action, if any, it is taking. This is necessary to prevent a HFM application from also trying to rectify the very same condition. An HFM application should avoid a interfering with ongoing corrective action taken by the device or application itself.

The corrective action may be a process, like hardware diagnostics or volume rebuild. In which case, the above requirement is fulfilled by expressing the instances representing the process.

The corrective action may be a state change, like reboot. In which case, the above requirement is fulfilled by expressing the state change in some CIM Instances.

In all cases, the profile or package that includes this package defines the standard events included and the associated, possible corrective actions taken in response to these events.

EXPERIMENTAL

22.2.8 RelatedElementCausingError

This package provides a mechanism in which the effect of a component failure on other components can be reported. the RelatedElementCausingError association defines what components are causing a particular component to failure or become degraded.

Some effects are more germane to the failure or degradation than others. In other words, there are primary and second effects. This association provides a mechanism for ranking the effect. The implementing shall provide the EffectCorrelation property, but it recommended that the implementation also provide the FailureRelationshipInitiated and Ranking properties

If there are these cause and effect relationships, the RelatedElementCausingError association should be implemented to report the causes of the failure or degradation.

22.2.9 HealthState

The HealthState property in LogicalDevice defines the state for a particular component. The OperationalStatus defines operational status. For example, a disk or port may be taken off-line for service. The component's health may still be OK or not OK. The two properties, when used in combination, disambiguate the health of the component. For example, an OperationStatus of 10 "Stopped" and a HealthState of 30 "Major Failure" means that the component is off-line and has failed. While an OperationalStatus of 10 "Stopped" and a HealthState of 5 "OK" for the very same component means that although the component is off-line, the component is still in good working order.

The HealthState of a component should not represent the health of any other component as well by way of a summary or aggregate health state. However, if the component is itself relies on other components for its health, because the component itself is an aggregate of components, then the HealthState may represent a summary HealthState by side-effect.

HealthState is a mandatory for all system device logical devices that are defined by the profile that includes this package. It is recommended that HealthState is something other than 0 "Unknown". However, a component may report "Unknown" after it has reported one of the other HealthStates. When HealthState changes from 5 "OK", it is mandatory that a LogicalDevice report some other HealthState (e.g., 30 "Major Failure") before reporting 0 "Unknown". Such a requirement is necessary, so that the client can notice the adverse state change via polling or indication before the component is no longer responding.

22.3 Health and Fault Management Considerations

Not defined in this standard.

22.4 Cascading Considerations

Not defined in this standard.

22.5 Use Cases

Not defined in this version of the standard.

22.6 CIM Elements

Table 226 describes the CIM elements for Health.

Table 226 - CIM Elements for Health

Element Name	Requirement	Description
22.6.1 CIM_ComputerSystem	Mandatory	
22.6.2 CIM_LogicalDevice	Mandatory	
22.6.3 CIM_RelatedElementCausingError	Optional	
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ComputerSystem AND SourceInstance.CIM_ComputerSystem::OperationalStatus[*] <> PreviousInstance.CIM_ComputerSystem::OperationalStatus[*]	Mandatory	CQL -Operational Status change of the device and application.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_LogicalDevice AND SourceInstance.CIM_LogicalDevice::HealthState <> PreviousInstance.CIM_LogicalDevice::HealthState	Mandatory	CQL -Health State change of the logical component.

22.6.1 CIM_ComputerSystem

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 227 describes class CIM_ComputerSystem.

Table 227 - SMI Referenced Properties/Methods for CIM_ComputerSystem

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	
Name		Mandatory	
OperationalStatus		Mandatory	Overall status of the Host.

22.6.2 CIM_LogicalDevice

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 228 describes class CIM_LogicalDevice.

Table 228 - SMI Referenced Properties/Methods for CIM_LogicalDevice

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	
HealthState		Mandatory	Reports the health of the component beyond the operational status.

22.6.3 CIM_RelatedElementCausingError

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 229 describes class CIM_RelatedElementCausingError.

Table 229 - SMI Referenced Properties/Methods for CIM_RelatedElementCausingError

Properties	Flags	Requirement	Description & Notes
FailureRelationshipInitiated		Optional	Reports the date and time when this cause and effect was created. The population of this property is RECOMMENDED.
EffectCorrelation		Mandatory	Describes the general nature of the cause and effect correlation.
Ranking		Optional	Describes the order of effect from 1, the highest effect, on. If there is only one of these associations between two elements, the ranking shall 1. Once more associations are added, then it RECOMMENDED that the implementation assist the client by stating which of the cause and effect relationship should be reviewed and addressed first. This property assists a client in accomplishing a triage of known problems.
Antecedent		Mandatory	Element causing the failure.
Dependent		Mandatory	

STABLE

STABLE

23 Job Control Profile

23.1 Synopsis

Profile Name: Job Control (Component Profile)

Version: 1.5.0

Organization: SNIA

Central Class: Service (e.g., StorageConfigurationService)

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

23.2 Description

In some profiles, some or all of the methods described may take some time to execute (longer than a HTTP time-out). In this case, a mechanism is needed to handle asynchronous execution of the method as a 'Job'.

NOTE The profile describes a specific use of the constructs and properties involved. The actual CIM capability may be more, but this specification clearly states what clients may depend on in SNIA profiles that implement the Job Control Profile.

23.2.1 Instance Diagram

A normal instance diagram is provided in Figure 33: "Job Control Profile Model".

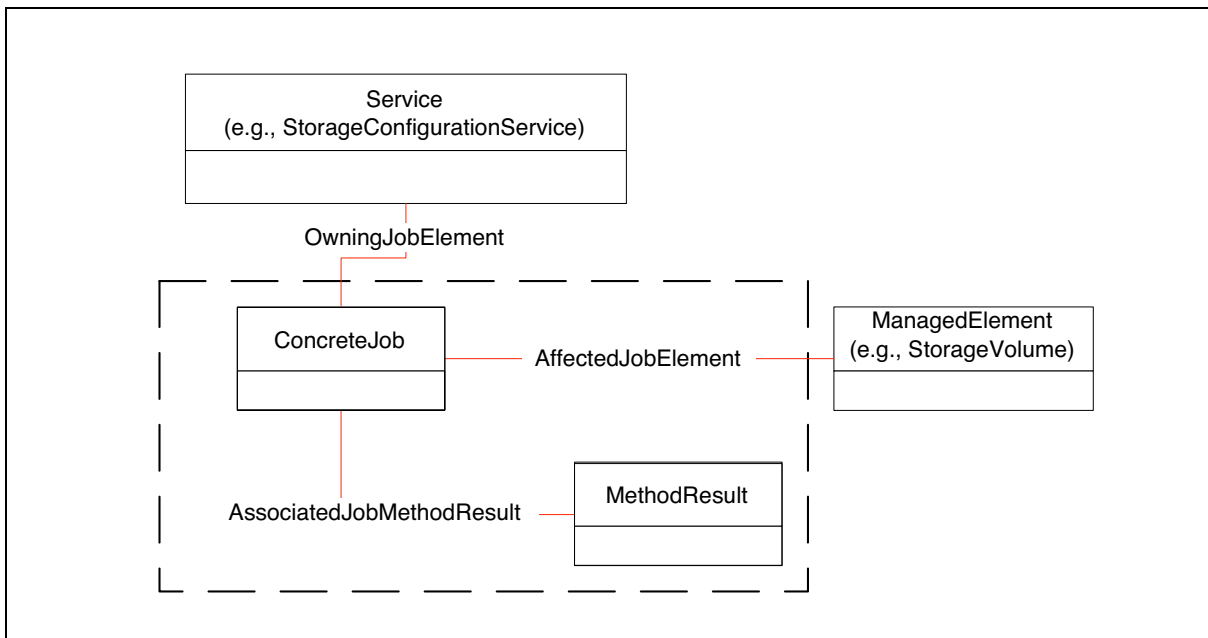


Figure 33 - Job Control Profile Model

When the Job Control Profile is implemented and a client executes a method that executes asynchronously, a reference to an instance of ConcreteJob is returned and the return value for the method is set to "Method parameters checked - job started".

The `ConcreteJob` instance allows the progress of the method to be checked, and instance Indications can be used to subscribe for Job completion.

The associations `OwningJobElement` and `AffectedJobElement` are used to indicate the service whose method created the job by side-effect and the element being affected by the job. The job itself may create, modify and/or delete many elements during its execution. The nature of this affect is the creation or deletion of the instances or associations or the modification of instance properties. These elements, albeit regular instances or associations, are said to be *affected* by the job. The elements linked by `AffectedJobElement` may change through the execution of the job, and in addition, the job may be associated to more than one Input and/or Output elements or other elements affected by side-effect. Input and Output elements are those referenced by method parameters of the same type, input and output parameters respectively.

EXPERIMENTAL

The following set of rules defines the nature of the `AffectedJobElement` associations for a given job in terms of the references passed as parameters to the service method that spawned the job. Obviously, the distinction of Input element from Output element in the following rules only makes sense if these parameters are not both Input and Output elements.

- If all Elements created by the method exists immediately upon the return from the method, then `AffectedJobElement` shall reference the Output Element.
- If the Output Element, one or more, does not exist until the job has completed, the `AffectedJobElement` shall reference the Input Element until the job completes, at which time `AffectedJobElement` shall then reference the Output Element instead.
- In the event the job fails and the Output Element created during the job and referenced by `AffectedJobElement` is no longer available, `AffectedJobElement` shall revert to referencing the Input Element.
- If the method affects elements without referencing elements as Output parameters, then the `AffectedJobElement` Association shall reference the Input element, one or more.
- If the method only modifies the elements referenced with method parameters, then the `AffectedJobElement` association references the modified elements. Elements modified by the job shall be reference by this association.
- If the method affects elements but references no elements as either Input or Output parameters or the only Input elements referenced are those of the elements to be deleted, then `AffectJobElement` associations shall exist to other elements that are affected by the job.
- Other elements whose references are not used in the method invocation, but that are created or modified by side-effect of the job's execution shall be associated to the job via the `AffectJobElement` association, but may cease to be associated once the job has finished execution.

The lifetime of a completed job instance, and thus the `AffectedJobElement` association to the appropriate Element is currently implementation dependent. However, the set of `AffectedJobElement` associations to Input and Output element present when the job finishes execution shall remain until the job is deleted.

23.2.2 MethodResult

Jobs are produced by side effect of the invocation of an extrinsic method. Reporting the resulting Job is the purpose of this component profile. The `MethodResult` class is used to report the extrinsic method called and the parameters passed to the method. In this way, third party observers of a CIMOM can tell what the job is and what it is doing. A `MethodResult` instance contains the `LifeCycle` indications that have been or would have been produced as the result of the extrinsic method invocation. That is, the instance

contains the indications whether or not there were the appropriate indication subscription at the time the indication were produced.

A client may fetch the method lifecycle indication produced when the method was called from the PreCallIndication attribute. This indication, an instance of InstMethodCall, contains the input parameters provided by the client that called the method.

A client may fetch the method lifecycle indication produced once the method execution was completed from the PostCallIndication. This indication contains the input parameters provided by the client that called the method and output parameters returned by the method implementation. Parameters that are both input and output parameters will contain the output parameter provided by the method implementation.

EXPERIMENTAL

23.2.3 OperationalStatus for Jobs

The OperationalStatus property is used to communicate that status of the job that is created. As such, it is critical that implementations are consistent in how this property is set. The values that shall be supported consistently are:

- 2 “OK” - combined with 17 “Completed” to indicate that the job completed with no error.
- 6 “Error” - combined with 17 “Completed” to indicate that the job did not complete normally and that an error occurred.
- 10 “Stopped” implies a clean and orderly stop.
- 17 “Completed” indicates the Job has completed its operation. This value should be combined with either 2 “OK” or 6 “Error, so that a client can tell if the complete operation passed (Completed with OK), and failure (Completed with Error).

23.2.4 JobState for Jobs

The JobState property is used to communicate Job specific states and statuses.

- 2 “New” - Job was created but has not yet started
- 3 “Starting” - Job has started
- 4 “Running” - Job is current executing
- 5 “Suspended” - Job has been suspended. The Job may be suspended for many reasons like it has been usurped by a higher priority or a client has suspended it (not described within this component profile).
- 6 “Shutting Down” - Job is completing its work, has been terminated, or has been killed. The Job may be cleaning up after only having completed some of its work.
- 7 “Completed” - Job has completed normally, its work has been completed successfully.
- 8 “Terminated” - Job has been terminated
- 9 “Killed” - Job has been aborted. The Job may not cleanup after itself.
- 10 “Exception” - Job failed and is in some abnormal state. The client may fetch the error conditions from the job. See 23.5.2.

Table 230 maps the standard mapping between the OperationalStatus and JobState properties on ConcreteJob. The actual values of the properties are listed in Table 230 with the associated value from the property's ValueMap qualifier.

Table 230 - OperationalStatus to Job State Mapping

OperationalStatus	JobState	Job is
2 "OK", 17 "Completed"	7 "Completed"	Completed normally
6 "Error", 17 "Completed"	10 "Exception"	Completed abnormally
10 "Stopped"	8 "Terminated"	Terminated
6 "Error"	9 "Killed"	Aborted / Killed
2 "OK"	4 "Running"	Executing
15 "Dormant"	2 "New"	Created but not yet executing
2 "OK", 8 "Starting"	3 "Starting"	Starting up
2 "OK"	5 "Suspended"	Suspended
2 "OK", 9 "Stopping"	6 "Shutting Down"	Terminated and potentially cleaning up
6 "Error"	6 "Shutting Down"	Killed and is aborting

23.2.5 Determining How Long a Job Remains after Execution

The Job shall report how long it will remain after it has finished executing, fails on its own, is terminated, or is killed. The TimeBeforeRemoval attribute reports a datetime offset.

The TimeBeforeRemoval and DeleteOnCompletion attributes are related. If the DeleteOnCompletion is FALSE, then the Job shall remain until is it explicitly deleted. If the DeleteOnCompletion is TRUE, then the Job shall exist for the length of time specified in the TimeBeforeRemoval attribute. An implementation may not support the setting of the DeleteOnCompletion attribute because it does not support the client modifying the Job instance.

The amount of time specified in the TimeBeforeRemoval should be five or more minutes. This amount of time allows a client to recognize that the Job has failed and retrieve the Error.

23.3 Health and Fault Management

The implementation should report CIM Errors from the ConcreteJob.GetError() method. See 22 Health Package for details.

EXPERIMENTAL

The standards messages specific to this profile are listed in Table 231. See 8 Standard Messages in *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5* for a description of standard messages and the list all standard messages

Table 231 - Standard Message for Job Control Profile

Message ID	Message Name
DRM22	Job failed to start
DRM23	Job was halted

EXPERIMENTAL

23.4 Cascading Considerations

Not defined in this standard.

23.5 Methods of the Profile

23.5.1 Job Modification

A Job instance may be modified. The DeleteOnCompletion and TimeBeforeRemoval properties are writable. If the intrinsic ModifyInstance method is supported, then the setting of both attributes shall be supported.

EXPERIMENTAL

23.5.2 Getting Error Conditions from Jobs

```
uint32 GetError(
    [Out, EmbeddedObject] string Error);
```

This method is used to fetch the reason for the job failure. The type of failure being report is when a Job stops executing on its own. That is, the Job was not killed or terminated. An Embedded Object, encoded in a string, shall returned if the method is both supported and the job has failed. The Job shall report the 10 "Exception" status when the Job has failed on its own.

The GetError method should be supported.

The Error string contains a Error instance. See 22 Health Package for details on how to process this CIM Instance.

EXPERIMENTAL

23.5.3 Suspending, Killing or Terminating a Job

A Job may be suspended, terminated or killed. Suspending a Job means that the Job will not be executing and be suspended until it is resumed. Terminating a job means to request that the Job stop executing and that the Job clean-up its state prior to completing. Killing a job means to request that the Job abort executing, usually meaning there is little or no clean-up of Job state.

```
uint32 RequestStateChange(
    [In] RequestedState,
    [In] TimeoutPeriod);
```

A client may request a state change on the Job.

- RequestedState - The standard states that can be requested are "Start", "Suspend", "Terminate", "Kill", "Service". A new Job may be started. A suspended Job may be resumed, using the "Started" requested status. A executing Job may be suspended, terminated, or killed. A new or executing Job may be put into the "Service" state. The "Service" state is vendor specific. An implementation can indicate what state transitions are supported by not returning the 4 098 "Invalid State Transition" return code
- TimeoutPeriod - The client expects the state transition to occur within the specified amount of time. The implementation may support the method but not this parameter.

Return codes:

- 0 "Completed with No Error"
- 1 "Not Supported" - The method is not supported
- 2 "Unknown/UnSpecified Error" - Failure for some vendor specific reason
- 3 "Can not complete within Timeout Period" - The requested amount of time is less than how long the requested state transition takes
- 4 "Failed"
- 5 "Invalid Parameters" - The parameters are incorrect
- 6 "In Use" - Another client has requested a state change that has not completed
- 4 096 "Method Parameters Checked - Transition Started" - The method can return before the state transition completes. This error code tells that calling that this situation has occurred
- 4 097 "Invalid State Transition" - The state change requested is invalid for the current state. 4 098 "Use of Timeout Parameter Not Supported" - This implementation does not support the TimeoutPeriod parameter. A client may pass a NULL for the TimeoutPeriod and try again. There is no mechanism to determine what state changes are supported by a particular implementation. Such a mechanism is planned for a future version of this specification.
- 4 099 "Busy" - A state change is underway in the Job and, as such, the state can not be changed. An implementation may use this return code to indicate the job can not be suspended, killed, or terminated at all or in the current phase of execution

23.6 Client Considerations and Recipes

23.6.1 Client Considerations

If the operation will take a while (longer than an HTTP timeout), a handle to a newly minted ConcreteJob is returned. This allows the job to continue in the background. Note a few things:

- The job is associated to the Service via OwingJobElement and is also linked to the object being modified/ created via AffectedJobElement. For example, a job to create a StorageVolume may start off pointing to a Pool until the Volume is instantiated at which point the association would change to the StorageVolume.
- These jobs do not have to get instantiated! If the method completes quickly, a null can be returned as a handle, as illustrated in Figure 34: "Storage Configuration".
- It may take some time before the Job starts.
- A Job may be terminated or killed.
- Jobs may be modified.
- Jobs may be restarted.

23.6.2 Recipes

No recipes are defined in this version of the standard.

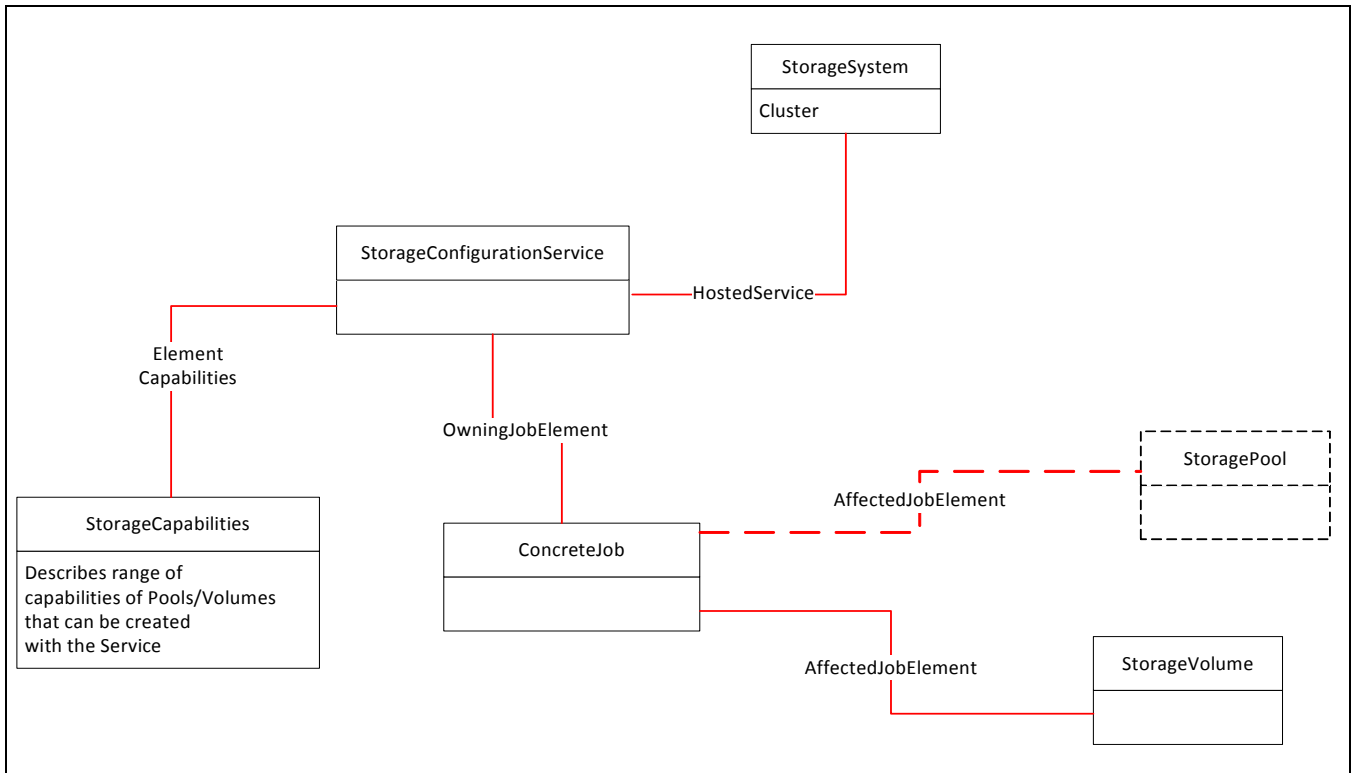


Figure 34 - Storage Configuration

23.7 Registered Name and Version

Job Control version 1.5.0 (Component Profile)

23.8 CIM Elements

Table 232 describes the CIM elements for Job Control.

Table 232 - CIM Elements for Job Control

Element Name	Requirement	Description
23.8.1 CIM_AffectedJobElement	Mandatory	
23.8.2 CIM_AssociatedJobMethodResult	Mandatory	
23.8.3 CIM_ConcreteJob	Mandatory	
23.8.4 CIM_MethodResult	Mandatory	
23.8.5 CIM_OwningJobElement	Mandatory	
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ConcreteJob AND SourceInstance.CIM_ConcreteJob::JobStatus <> PreviousInstance.CIM_ConcreteJob::JobStatus	Optional	CQL -Deprecated. Modification of Job Status for a Concrete Job.

Table 232 - CIM Elements for Job Control

Element Name	Requirement	Description
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ConcreteJob AND SourceInstance.CIM_ConcreteJob::PercentComplete <> PreviousInstance.CIM_ConcreteJob::PercentComplete	Mandatory	CQL -Modification of Percentage Complete for a Concrete Job.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ConcreteJob AND ANY SourceInstance.CIM_ConcreteJob::OperationalStatus[*] = 17 AND ANY SourceInstance.CIM_ConcreteJob::OperationalStatus[*] = 2	Mandatory	CQL -Modification of Operational Status for a Concrete Job to 'Complete' and 'OK'.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ConcreteJob AND ANY SourceInstance.CIM_ConcreteJob::OperationalStatus[*] = 17 AND ANY SourceInstance.CIM_ConcreteJob::OperationalStatus[*] = 6	Mandatory	CQL -Modification of Operational Status for a Concrete Job to 'Complete' and 'Error'.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ConcreteJob AND SourceInstance.CIM_ConcreteJob::JobState <> PreviousInstance.CIM_ConcreteJob::JobState	Mandatory	CQL -Modification of Job State for a Concrete Job.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_ConcreteJob	Mandatory	Creation of a ConcreteJob.

23.8.1 CIM_AffectedJobElement

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 233 describes class CIM_AffectedJobElement.

Table 233 - SMI Referenced Properties/Methods for CIM_AffectedJobElement

Properties	Flags	Requirement	Description & Notes
AffectedElement		Mandatory	The ManagedElement affected by the execution of the Job.
AffectingElement		Mandatory	The Job that is affecting the ManagedElement.

23.8.2 CIM_AssociatedJobMethodResult

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 234 describes class CIM_AssociatedJobMethodResult.

Table 234 - SMI Referenced Properties/Methods for CIM_AssociatedJobMethodResult

Properties	Flags	Requirement	Description & Notes
Job		Mandatory	The Job that has parameters.
JobParameters		Mandatory	The parameters for the method which by side-effect created the Job.

23.8.3 CIM_ConcreteJob

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 235 describes class CIM_ConcreteJob.

Table 235 - SMI Referenced Properties/Methods for CIM_ConcreteJob

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
Name		Mandatory	The user-friendly name for this instance of Job. In addition, the user-friendly name can be used as a property for a search or query. (Note: Name does not have to be unique within a namespace.)".
OperationalStatus		Mandatory	Describes whether the Job is running or not.
JobStatus		Optional	Add additional detail beyond OperationalStatus about the runtime status of the Job. This property is vendor specific.
JobState		Mandatory	Add additional detail beyond the OperationalStatus about the runtime state of the Job.
ElapsedTime		Optional	The time interval that the Job has been executing or the total execution time if the Job is complete.
PercentComplete		Mandatory	The percentage of the job that has completed at the time that this value is requested. Optimally, the percentage should reflect the amount of work accomplished in relation to the amount of work left to be done. 0 percent complete means that the job has not started and 100 percent complete means the job has finished all its work. However, in the degenerate case, 50 percent complete means that the job is running and may remain that way until the job completes.
DeleteOnCompletion		Mandatory	Indicates whether or not the job should be automatically deleted upon completion. If this property is set to false and the job completes, then the extrinsic method DeleteInstance shall be used to delete the job versus updating this property. Even if the Job is set to delete on completion, the job shall remain for some period of time, see GetError() method.
ErrorCode		Optional	A vendor specific error code. This is set to zero if the job completed without error.
ErrorDescription		Optional	A free form string containing the vendor error description.
TimeBeforeRemoval		Mandatory	The amount of time the job will exist after the execution of the Job if DeleteOnCompletion is set to FALSE. Jobs that complete successfully or fail shall remaining for at least this period of time before being removed from the model (CIMOM).

Table 235 - SMI Referenced Properties/Methods for CIM_ConcreteJob

Properties	Flags	Requirement	Description & Notes
GetError()		Mandatory	This method is used to retrieve the error that caused the Job to fail. The Job shall remain in the model long enough to allow client to a) notice that the job was stopped executing and b) to retrieve the error using this method. There are not requirements for how long the job must remain; however, it is suggested that the Job remain for at least five minutes. JobStatus=10 (Exception) tell the client that the job failed and this method can be called to retrieve the reason why embedded in the CIM_Error, see GetError() method.
RequestStateChange()		Optional	This method changes the state of the job. The client may suspend, terminate, or shutdown the job. To terminate a job means to request a clean shutdown of the job, have it finish some portion of it's work and terminate or to roll back the changes done by the job to date. The implement can make the choice which behavior. To kill a job means to abort the job, perhaps leaving some element of the work partially done and in an unknown state.

23.8.4 CIM_MethodResult

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 236 describes class CIM_MethodResult.

Table 236 - SMI Referenced Properties/Methods for CIM_MethodResult

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
PreCallIndication		Mandatory	Contains a copy of the CIM_InstMethodCall produced when the configuration or control change method was called. This Embedded Instance shall contain the configuration or control change extrinsic method name (MethodName) and parameters (MethodParameters).
PostCallIndication		Mandatory	Contains a copy of the CIM_InstMethodCall produced when the configuration or control change method has completed execution and control was returned to the client. This Embedded Instance shall contain the configuration or control change extrinsic method name (MethodName) and parameters (MethodParameters).

23.8.5 CIM_OwningJobElement

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 237 describes class CIM_OwningJobElement.

Table 237 - SMI Referenced Properties/Methods for CIM_OwningJobElement

Properties	Flags	Requirement	Description & Notes
OwningElement		Mandatory	The ManagedElement responsible for the creation of the Job. (e.g., StorageConfigurationService).
OwnedElement		Mandatory	The Job created by the ManagedElement.

STABLE

Job Control Profile

STABLE**24 Location Profile****24.1 Synopsis**

Profile Name: Location (Component Profile)

Version: 1.4.0

Organization: SNIA

Central Class: Location

Scoping Class: PhysicalPackage

Related Profiles: Not defined in this standard.

24.2 Description

Associated with product information, a PhysicalPackage may also have a location. This is indicated using an instance of a Location class and the PhysicalElementLocation association.

24.2.1 Instance Diagram

Figure 35: "Location Instance" illustrates a typical instance diagram.

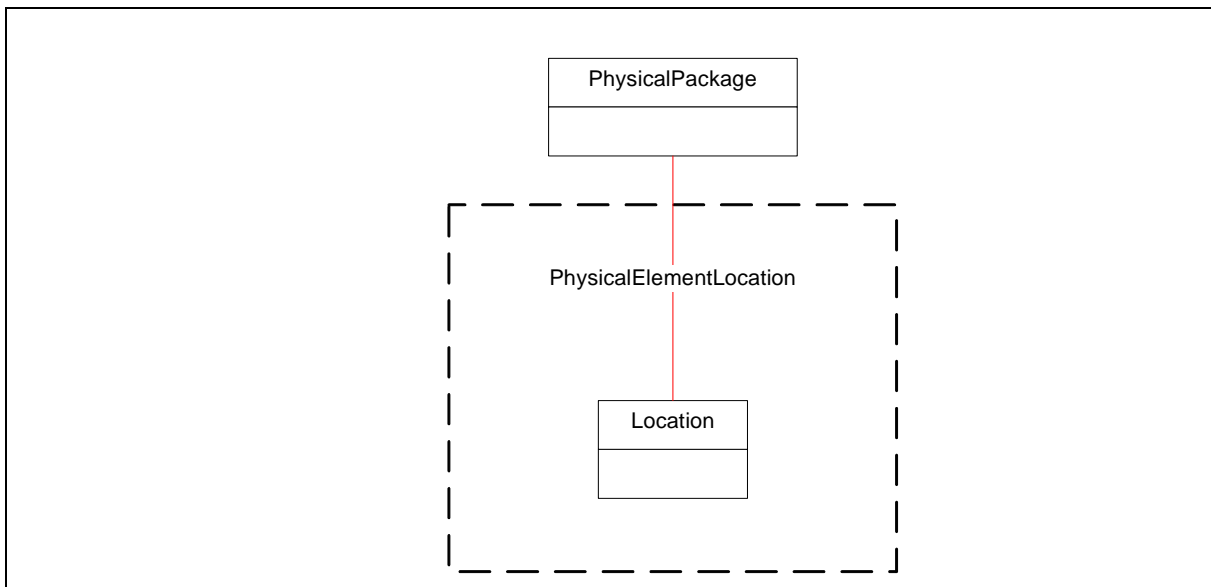


Figure 35 - Location Instance

24.3 Health and Fault Management Considerations

Not defined in this standard.

24.4 Cascading Considerations

Not defined in this standard.

24.5 Methods of the Profile

Not defined in this standard.

24.6 Use Cases

Not defined in this version of the standard.

24.7 CIM Elements

Table 238 describes the CIM elements for Location.

Table 238 - CIM Elements for Location

Element Name	Requirement	Description
24.7.1 CIM_Location	Mandatory	
24.7.2 CIM_PhysicalElementLocation	Mandatory	Associates the location to package.

24.7.1 CIM_Location

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 239 describes class CIM_Location.

Table 239 - SMI Referenced Properties/Methods for CIM_Location

Properties	Flags	Requirement	Description & Notes
Name		Mandatory	A free-form string defining a label for the Location.
PhysicalPosition		Mandatory	A free-form string indicating the placement of a PhysicalElement.
ElementName		Optional	User-friendly name.
Address		Optional	A free-form string indicating a street, building or other type of address for the PhysicalElement's Location.

24.7.2 CIM_PhysicalElementLocation

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 240 describes class CIM_PhysicalElementLocation.

Table 240 - SMI Referenced Properties/Methods for CIM_PhysicalElementLocation

Properties	Flags	Requirement	Description & Notes
Element		Mandatory	
PhysicalLocation		Mandatory	

STABLE

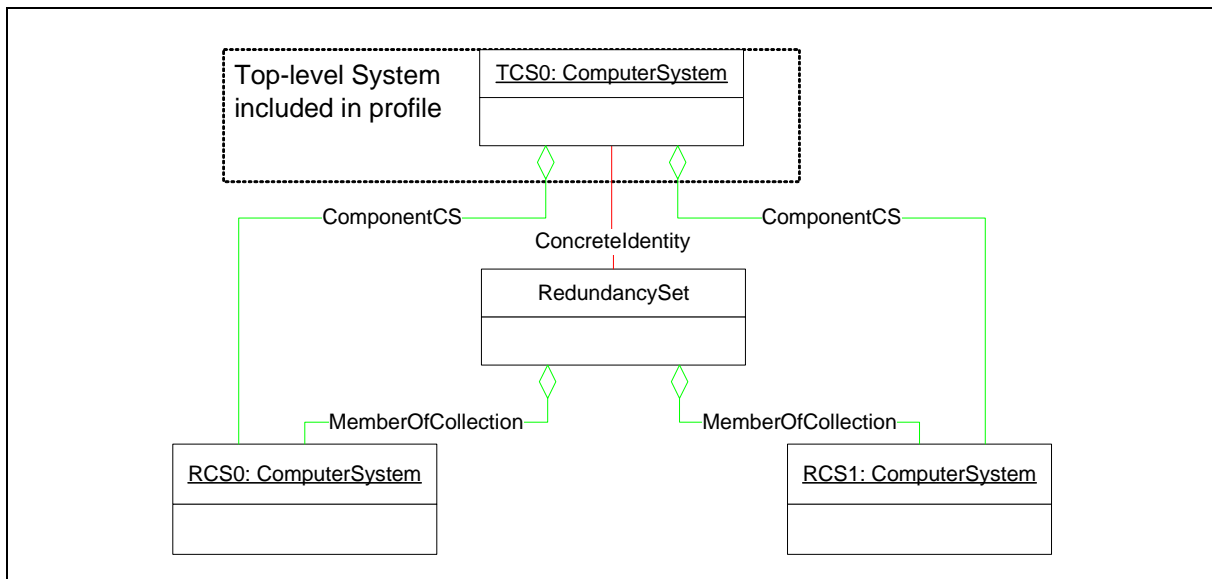
Location Profile

STABLE**25 Multiple Computer System Profile****25.1 Synopsis****Profile Name:** Multiple Computer System (Component Profile)**Version:** 1.2.0**Organization:** SNIA**Central Class:** ComputerSystem**Scoping Class:** a top-level ComputerSystem**Related Profiles:** Table 241 describes the related profiles for Multiple Computer System.**Table 241 - Related Profiles for Multiple Computer System**

Profile Name	Organization	Version	Requirement	Description
Storage Server Asymmetry	SNIA	1.4.0	Optional	Experimental.

25.2 Description

The Multiple Computer System Profile models multiple systems that cooperate to present a “virtual” computer system with additional capabilities or redundancy. This virtual aggregate system is sometimes referred to as a cluster, and is illustrated in Figure 36: “Two Redundant Systems Instance Diagram”.

**Figure 36 - Two Redundant Systems Instance Diagram**

The general pattern for the redundancy aspect of Multiple Systems uses an instance of RedundancySet to aggregate multiple “real” ComputerSystem instances (labeled RCS0 and RCS1 in the diagram). Another ComputerSystem instance (TCS0) is associated to the RedundancySet instance using a ConcreteIdentity association and is associated to the real ComputerSystems using ComponentCS.

25.2.1 Top Level System

The top (“virtual”) system in this diagram (labeled TCS0) is referred to as the Top Level System. Note that for single-system configurations, the top-level system is the only system. Top-level systems have characteristics different from the underlying ComputerSystem instances.

The Top Level System is associated to the registered profile described in 35 Server Profile. Other elements such as LogicalDevices (ports, volumes), ServiceAccessPoints, and Services are associated to the top-level system if these elements are supported by multiple underlying systems (for example, the underlying systems provide failover and/or load balancing). Alternatively, elements can be associated to an underlying system if that system is a single point of failure. For example, a RAID array may associate StorageVolume instances to a top-level system since these are available when one underlying system (RAID controller) fails, all the port elements are associated to one underlying system because the ports become unavailable when this system fails.

The Dedicated property is required for top-level systems. Each profile defines the values that are appropriate for Dedicated.

25.2.2 Non-Top-Level Systems

Each ComputerSystem instance shall have a unique Name property. For non-top-level systems, Name may be vendor-unique; in which case, NameFormat shall be set to “Other”.

ComputerSystem.Dedicated should not be used in non-top-level systems.

Non-top-level systems shall not be associated to registered profiles.

Each non-top-level ComputerSystem shall be associated to the top-level system using ComponentCS. Note that non-top-level systems may not be members of a RedundancySet. For example, a top-level system may be associated to a RedundancySet with two systems as described in Figure 36: "Two Redundant Systems Instance Diagram" and also associated via ComponentCS to another Computer (not a member of a RedundancySet) representing a service processor.

25.2.3 Types of RedundancySets

The TypeOfSet property of RedundancySet is a list describing the types of redundancy. Its values are summarized in Table 242.

Table 242 - Redundancy Type

Redundancy Type	Description
N+1	All ComputerSystems are active, are unaware and function independent of one another. However, there exists at least one extra ComputerSystem to achieve functionality.
Load Balanced	All computer systems are active. However, their functionality is not independent of each other. Their functioning is determined by some sort of load balancing algorithm (implemented in hardware and/or software). 'Sparing' is implied (i.e., each computer system can be a spare for the other(s)).
Sparing	All computer systems are active and are aware of each other. However, their functionality is independent until failover. Each computer system can be a spare for the other(s).
Limited Sparing	All members are active, and they may or may not be aware of each and they are not spares for each other. Instead, their redundancy is indicated by the IsSpare relationship.
Other/Unspecified	The relationship between the computer systems is not specified.

25.2.4 Multiple Tiers of Systems

The diagram above describes two tiers of systems; the real systems (labeled RCS0 and RCS1) in the lower tier are aggregated into a top-level system (TCS0) in the upper tier. There may be more than two tiers, as depicted in Figure 37: "Multiple Redundancy Tier Instance Diagram".

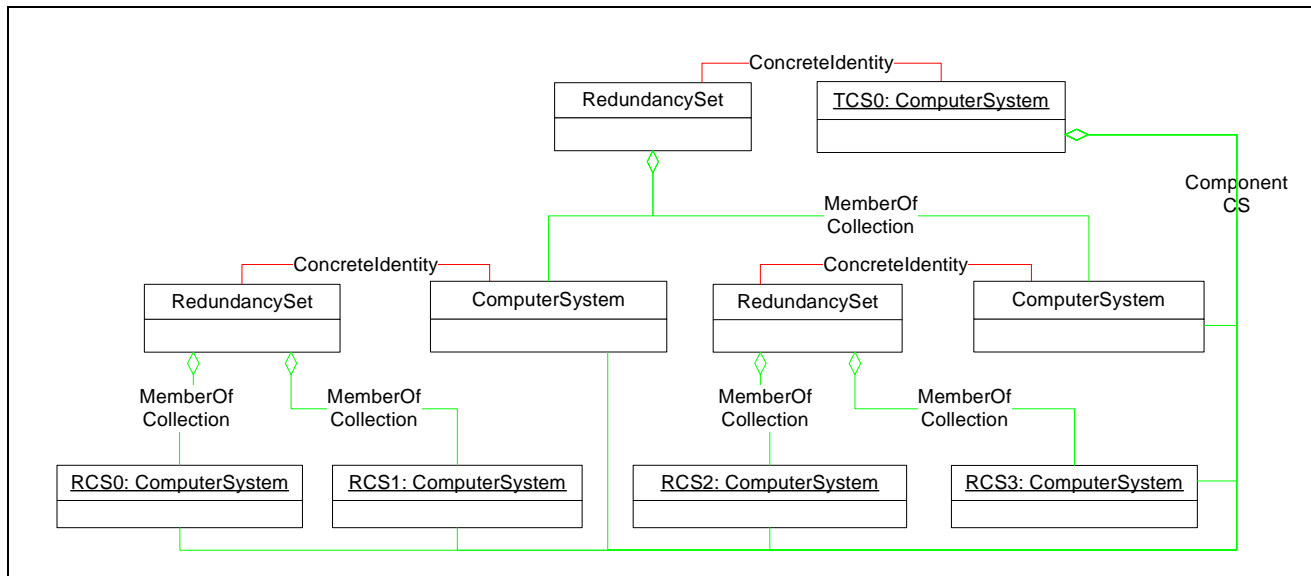


Figure 37 - Multiple Redundancy Tier Instance Diagram

The systems in the bottom tier (RCS0-RCS3) represent "real" systems.

RedundancySet.TypeOfSet can be used as part of multiple tier configurations to describe different types of redundancy at different tiers. For example, a virtualization system has four controllers that operate in pairwise redundancy. This could be modeled using the model in the diagram above and setting TypeOfSet in the top RedundancySet to "N+1" and setting TypeOfSet to "LoadBalancing" in the lower two RedundancySets.

25.2.5 Associations between ComputerSystems and other Logical Elements

SystemDevice associates device (subclasses of LogicalDevice such as LogicalPort or StorageVolume) and ComputerSystem instances. The cardinality of SystemDevice is one-to-many; a LogicalDevice may be associated with one and only one ComputerSystem. If the device availability is equivalent to that of the top-level system, it shall be associated to the top-level system via SystemDevice. If the device may become unavailable while the system as a whole remains available, the device shall be associated to a non-top-level system that has availability equivalent to the device. This system could be a real system or a system in an intermediate tier (representing some redundancy less than full redundancy).

This same approach shall be used for all other logical CIM elements with associations to systems. For example, HostedService and HostedAccessPoint shall associate elements (services, access points, and protocol endpoints) to the ComputerSystem with availability to the element.

Based on the arrangement of systems in figure 31, associations from systems to service and capabilities classes shall not be lower than associations to other classes. For the purpose of formally stating this rule, each ComputerSystem is assigned a level number. The profile's top-level ComputerSystem has level number 0. The ComputerSystem instances that are members of RedundancySets associated via ConcretelDentity to the top-level system have level number 1. The members of redundancy sets associated to the level number 1 systems via ConcretelDentity have level number 2. In general, the ComputerSystem members of redundancy sets associated to the level number n systems via ConcretelDentity have level number n+1. The level of non-system objects is the level of the

ComputerSystem instance associated to the object via associations such as SystemDevice, HostedAccessPoint, HostedService, or ElementCapabilities.

Figure 38: "System Level Numbers" demonstrates these system level numbers using the same configuration from Figure 37: "Multiple Redundancy Tier Instance Diagram". Note that ComponentCS diagrams are omitted from this diagram to avoid clutter.

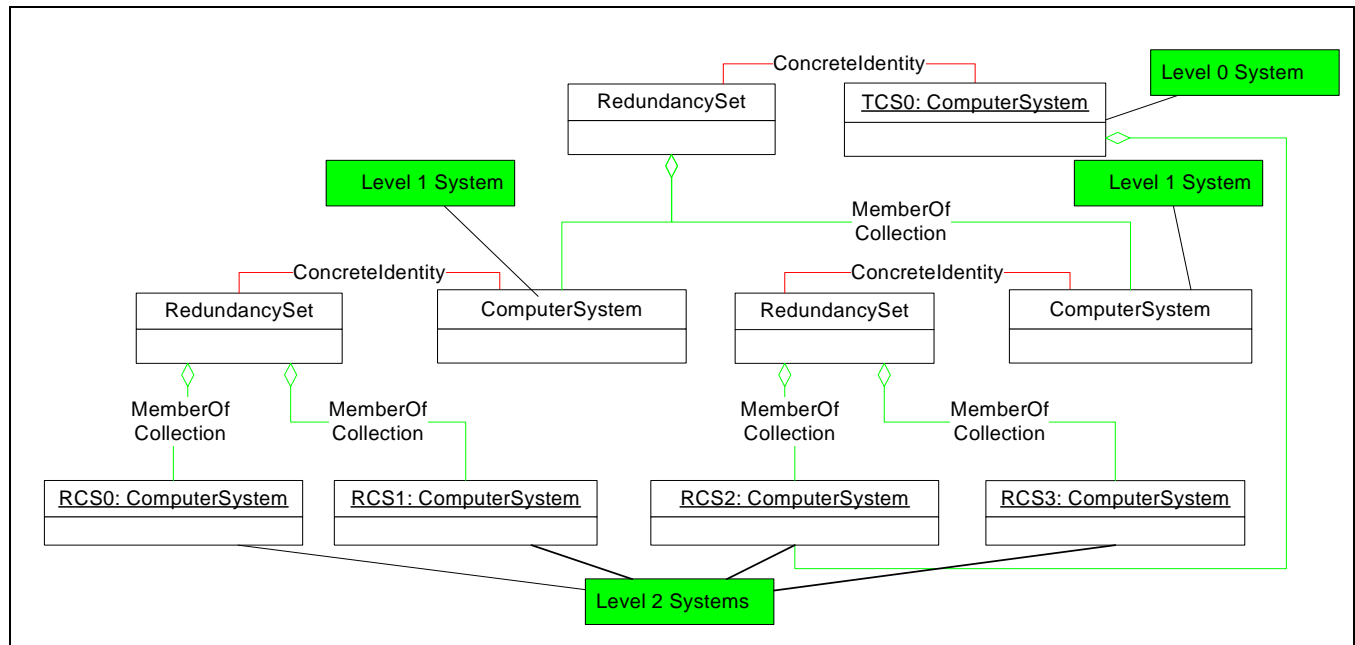


Figure 38 - System Level Numbers

All subclasses of CIM_Service and CIM_Capabilities shall have a level number less than or equal to the level number of storage classes (ports, volumes, etc.) that are influenced by the properties and methods of the Service and Capabilities classes. In some cases, different storage classes are influenced by different Service or Capabilities classes; the "level number less than or equal to" requirement may apply differently to different Service/Capabilities classes. It is always valid to associate Service and Capabilities classes to the top-level ComputerSystem since the level number of the top-level system (0) is always less than or equal to the level number of any other system.

Example 1 - An array with two controllers is modeled as a top-level ComputerSystem with real systems representing the controllers. The system's storage volumes remain available when one controller fails, but each LogicalPort becomes unavailable when a controller fails. The StorageVolumes should be associated to the top-level ComputerSystem and the LogicalPorts should be associated to one of the real ComputerSystems.

Example 2 - An array with four pair-wise redundant controllers. Each LogicalPort is associated with a pair of controllers - if one controller in a pair fails, the port is still accessible through the alternate controller. This corresponds to Figure 37: "Multiple Redundancy Tier Instance Diagram"; the ports should be associated with one of the ComputerSystems in the middle tier.

A provider shall delete and create associations between ComputerSystems and logical elements (e.g., ports, logical devices) during failover or failback to represent changes in availability. This includes SystemDevice, HostedAccessPoint, HostedService, or HostedFileSystem associations (and other associations weak to systems). The effect of the creation and deletion of associations is to switch these elements from one ComputerSystem to another. The profiles that include Multiple Computer System Profile shall specify the affected associations and indications for creation and deletion of these associations.

25.2.6 Associations between ComputerSystems and PhysicalPackages and Products

The relationship between ComputerSystems, PhysicalPackages, and Products is defined in the Physical Package Package (see 26 Physical Package Package) which may be required by the profile including the Multiple Computer System Profile. Typically, the top-level system is associated to a PhysicalPackage which is associated to a Product. Non-top-level systems may also be associated to PhysicalPackage and indirectly to a Product. If all underlying ComputerSystems share the same physical package, a single PhysicalPackage should be associated to the upper ComputerSystem.

The relationships between ComputerSystems, redundancy sets, and CIM logical elements serve as a redundancy topology - informing the client of the availability of subsets of logical elements. The relationships between PhysicalPackages and logical elements serve as a physical topology. These two topologies need not be equivalent. Consider these examples:

Example 1: a RAID array with a single controller (no redundancy); the controller and all backend disks are housed in a single chassis. This is modeled as a single ComputerSystem, no RedundancySets, no ComponentCS associations, and a single PhysicalPackage with a single associated Product.

Example 2: a RAID array with two redundant controllers; both controllers and all backend disks share a single chassis. In this case, the redundancy topology matches Figure 36: "Two Redundant Systems Instance Diagram". The top-level ComputerSystem is associated to a PhysicalPackage with a single associated Product.

Example 3: two arrays described in example 1 are assembled as part of common rack and sold as a single product. Note that although there are two controllers, there is no redundancy - the two controllers act completely independently. This is modeled as two top-level computer systems attached to separate PhysicalPackages (representing the two internal chassis); These two PhysicalPackages have a Container association to third PhysicalPackage representing the assembly - which has an association to a Product.

Example 4: two arrays described in Example 1 are assembled as part of a common rack and also share a high-speed trunk and a mutual failover capability. This failover capability means the two controllers share a RedundancySet and common top-level system. The result is similar to example 2, but each real ComputerSystem is now associated to separate PhysicalPackages which have Container associations to a common PhysicalPackage.

25.2.7 Storage Systems without Multiple Systems

In configurations where the instrumentation does not model multiple ComputerSystem instances, all the associations described above reference the one and only ComputerSystem.

25.2.8 Durable Names and Correlatable IDs of the Profile

This profile does not impose any requirements on names. The requirements for ComputerSystem names are defined in the profiles that depend on Multiple Computer System Profile and in *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5 7* Correlatable and Durable Names. Clients should not expect that a network name or IP address is exposed as a ComputerSystem property. The Access Points Profile should be used to model a network access point.

25.3 Health and Fault Management Considerations

The requirements for OperationalStatus of a ComputerSystem are discussed in 22 Health Package.

25.4 Cascading Considerations

Not defined in this standard

25.5 Methods of the Profile

This profile does not include any extrinsic methods. A client may use this profile to discover information about the topology of computer systems, but cannot change the topology.

25.6 Use Cases

25.6.1 Navigate the Redundancy Topology

A client cannot generally interoperably navigate the redundancy topology using ComponentCS because some Component CS associations may not parallel RedundancySet associations. But a client may use ComponentCS selectively to speed up certain tasks. In particular, a client may locate the top-level system from other ComputerSystems using ComponentCS.

25.6.2 Find Top-level Computer Systems

Top-level systems are the only objects in SMI-S associated to RegisteredProfile via ElementConformsToProfile.

25.7 CIM Elements

Table 243 describes the CIM elements for Multiple Computer System.

Table 243 - CIM Elements for Multiple Computer System

Element Name	Requirement	Description
25.7.1 CIM_ComponentCS	Mandatory	Associates non-top-level systems to the top-level system.
25.7.2 CIM_ComputerSystem (Non-Top-Level System)	Mandatory	Non-Top-level System.
25.7.3 CIM_ConcretelDentity	Mandatory	Associates aggregate (possibly top-level) ComputerSystem and RedundancySet.
25.7.4 CIM_IsSpare	Optional	Associates the ComputerSystem that may be used as a spare to the RedundancySet of ActiveComputerSystem.
25.7.5 CIM_MemberOfCollection	Mandatory	Associates RedundancySet and its member ComputerSystems.
25.7.6 CIM_RedundancySet	Mandatory	
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_ComputerSystem	Mandatory	Creation of a ComputerSystem instance.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_ComputerSystem	Mandatory	Deletion of a ComputerSystem instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_ComputerSystem AND SourceInstance.CIM_ComputerSystem::OperationalStatus <> PreviousInstance.CIM_ComputerSystem::OperationalStatus	Mandatory	CQL -Change of Operational Status of a ComputerSystem instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_RedundancySet AND SourceInstance.CIM_RedundancySet::RedundancyStatus <> PreviousInstance.CIM_RedundancySet::RedundancyStatus	Mandatory	CQL -Change of redundancy status.

25.7.1 CIM_ComponentCS

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 244 describes class CIM_ComponentCS.

Table 244 - SMI Referenced Properties/Methods for CIM_ComponentCS

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	The Top-Level ComputerSystem; must be associated to a RegisteredProfile.
PartComponent		Mandatory	The contained (Sub)ComputerSystem.

25.7.2 CIM_ComputerSystem (Non-Top-Level System)

Non-Top-level system.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 245 describes class CIM_ComputerSystem (Non-Top-Level System).

Table 245 - SMI Referenced Properties/Methods for CIM_ComputerSystem (Non-Top-Level System)

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	
Name		Mandatory	
NameFormat		Mandatory	Non-top-level system names are not correlatable, any format is valid.
ElementName		Mandatory	
OperationalStatus		Mandatory	

25.7.3 CIM_ConcretelDentity

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 246 describes class CIM_ConcretelDentity.

Table 246 - SMI Referenced Properties/Methods for CIM_ConcretelDentity

Properties	Flags	Requirement	Description & Notes
SystemElement		Mandatory	Reference to ComputerSystem
SameElement		Mandatory	Reference to RedundancySet

25.7.4 CIM_IsSpare

Associates the ComputerSystem that may be used as a spare to the RedundancySet of ActiveComputerSystem.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 247 describes class CIM_IsSpare.

Table 247 - SMI Referenced Properties/Methods for CIM_IsSpare

Properties	Flags	Requirement	Description & Notes
SpareStatus		Mandatory	
FailoverSupported		Mandatory	
Dependent		Mandatory	The RedundancySet.
Antecedent		Mandatory	The spare system.

25.7.5 CIM_MemberOfCollection

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 248 describes class CIM_MemberOfCollection.

Table 248 - SMI Referenced Properties/Methods for CIM_MemberOfCollection

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	Reference to RedundancySet
Member		Mandatory	Reference to ComputerSystem

25.7.6 CIM_RedundancySet

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 249 describes class CIM_RedundancySet.

Table 249 - SMI Referenced Properties/Methods for CIM_RedundancySet

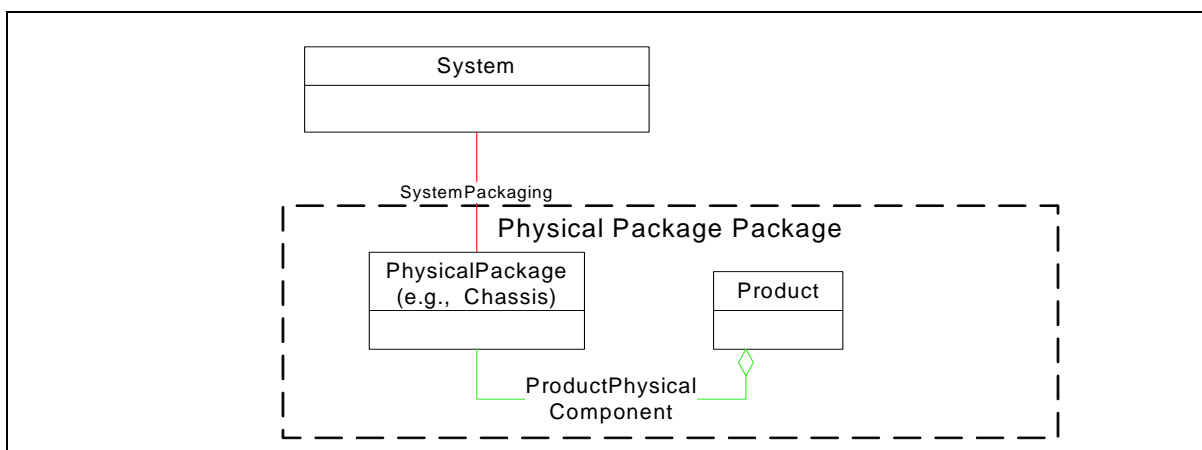
Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
RedundancyStatus		Mandatory	The redundancy status shall be either 'Unknown' 0, 'Redundant' 2, 'Degraded Redundancy' 3, 'Redundancy Lost' 4, or 'Overall Failure' 5. implementation should report 2 or 3 most of the time, although it may report 0. It should report 2 or 3 when there is at least one spare per the RedundancySet. It should report 4 when there are no more spares (via IsSpare association) per the RedundancySet. It should report 5 when there is a failure of the RedundancySet.
TypeOfSet		Mandatory	

STABLE

STABLE**26 Physical Package Package****26.1 Synopsis****Profile Name:** Physical Package (Component Profile)**Version:** 1.5.0**Organization:** SNIA**Central Class:** PhysicalPackage**Scoping Class:** a ComputerSystem in a separate autonomous profile**Related Profiles:** Not defined in this standard.**26.2 Description**

Physical Package Package models information about a storage system's physical package and optionally about internal sub-packages. A System is 'realized' using a SystemPackaging association to a PhysicalPackage (or a subclasses such as Chassis). The physical containment model can then be built up using Container associations and subclasses (such as PackageInChassis).

Physical elements are described as products using the Product class and ProductPhysicalComponent associations, as shown in Figure 39. The Product instances may be built up into a hierarchy using the ProductParentChild association. The Product class holds information such as vendor name, serial number and version.

**Figure 39 - Physical Package Package Mandatory Classes****26.2.1 Well Defined Subcomponents**

In addition to defining physical packages at the "System" level, PhysicalPackage may also be defined at a lower, subcomponent level. For example, PhysicalPackage is used in the Disk Drive Lite Profile and for devices supported by storage media libraries (e.g., TapeDrive and ChangerDevice). If the subcomponents are supported by the Profile, they should model their physical packaging. When subcomponents are modeled, there shall be a container relationship between their physical package and the containing package (e.g., the System level physical package). In addition, there shall be a ProductParentChild association between the subcomponent Product and the parent Product.

The Physical Package constructs may also be used to model other aspects of the environment. However, this is not mandatory. Note that each logical device may be realized by a card. The cards are contained in a controller chassis.

When establishing physical packages for logical device subcomponents (e.g., disk drives, changers, etc.) the provider shall populate both Container and Realizes associations. When establishing physical packages for processor subcomponents

(e.g., non-top level systems.) the provider shall populate both Container and SystemPackaging associations. When establishing the Product instances for the subcomponent packages the provider shall populate the ProductParentChild association to the parent product. This is illustrated in Figure 47.

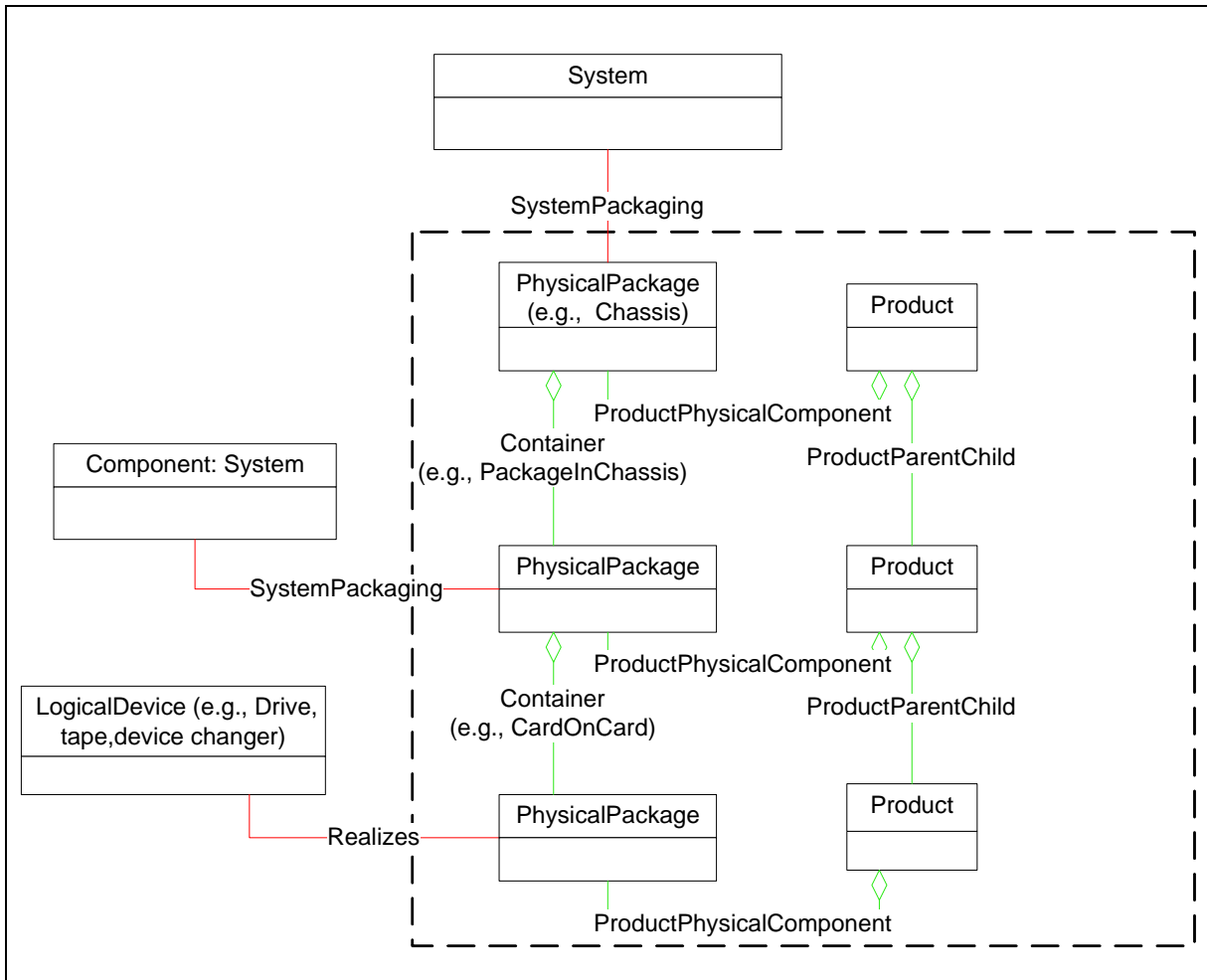


Figure 40 - Modeling for well defined subcomponents

26.2.2 Multiple Product Identities

Instrumentation may optionally describe multiple product identities for a physical package, for example, product information for both an OEM and vendor. This information should be modeled as multiple instances of CIM_Product associated with the LogicalIdentity association. The Product instance that clients should treat as primary is directly associated with PhysicalPackage via ProductPhysicalComponent. Additional product instances are associated with the primary product using the LogicalIdentity association.

Figure 41: "Physical Package Package with Optional Classes" shows an example of the use of mandatory and optional physical package classes.

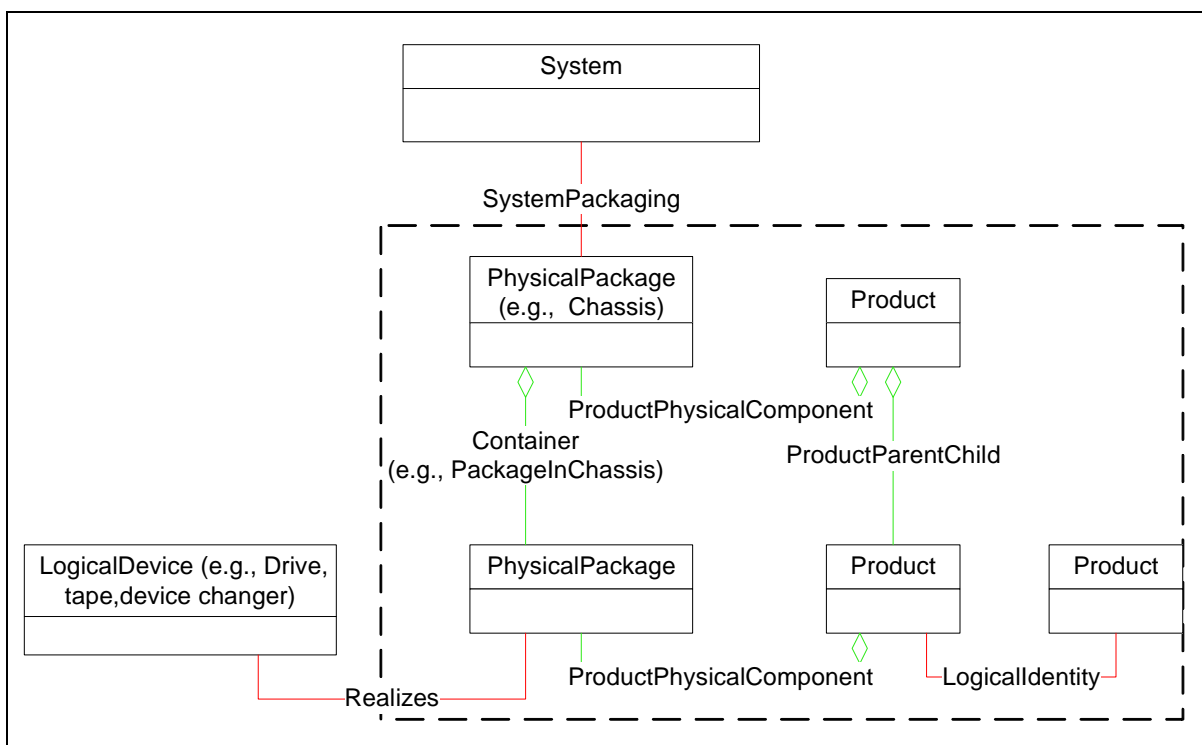


Figure 41 - Physical Package Package with Optional Classes

26.3 Health and Fault Management Considerations

Not defined in this standard.

26.4 Cascading Considerations

Not defined in this standard.

26.5 Methods of this Profile

Not defined in this standard.

26.6 Use Cases

26.6.1 Find Asset Information

Information about a system is modeled in PhysicalPackage. PhysicalPackage may be subclassed to Chassis; the more general PhysicalPackage is used here to accommodate device implementations that are deployed in multiple chassis. PhysicalPackage has an associated Product with physical asset information such as Vendor and Version.

26.6.2 Finding Product information

To locate product information (Vendor, Serial number and product versions) information about a device that conforms to the profile, you would start with the "top-level" computer system and traverse the SystemPackaging to the PhysicalPackage (e.g., a Chassis). From the PhysicalPackage, the client would then traverse the ProductPhysicalComponent association to locate the Product instance. The primary Vendor, Serial Number and version for the device is in the Product instance associated with the

PhysicalPackage. Additional product identities may be associated with the primary Product using the LogicalIdentity association.

26.6.3 Finding Asset information

There are certain subcomponents of a device that a client may be interested in locating. For example, disk drives in an array or changer devices in a library. To locate the asset information of these subcomponents, the client would follow the ProductParentChild association from the system Product to lower level Products.

Alternatively, if the client is starting from a LogicalDevice, it can locate the PhysicalPackage by following the Realizes association from the LogicalDevice. From the PhysicalPackage, the client can find the Product information by traversing the ProductPhysicalComponent association.

26.7 CIM Elements

Table 250 describes the CIM elements for Physical Package.

Table 250 - CIM Elements for Physical Package

Element Name	Requirement	Description
26.7.1 CIM_Container	Optional	Associates a PhysicalPackage to its component physical packages (e.g., Drives in a Storage System).
26.7.2 CIM_LogicalIdentity	Optional	Associates the primary product information to secondary product information.
26.7.3 CIM_PhysicalElementLocation	Conditional	Conditional requirement: Support for the Location profile. Associates the physical package of the system to its location.
26.7.4 CIM_PhysicalPackage (Component)	Optional	A physical package for a component of the overall system.
26.7.5 CIM_PhysicalPackage (System)	Mandatory	The physical package for the overall system.
26.7.6 CIM_Product (Component)	Optional	The product information for a physical package that is a component of the system.
26.7.7 CIM_Product (System)	Mandatory	The product information for the physical package of the system.
26.7.8 CIM_ProductParentChild	Optional	If more than one product comprises a system, this association should be used to indicate the 'parent' product.
26.7.9 CIM_ProductPhysicalComponent (Component)	Optional	Associates a component physical package to its product information.
26.7.10 CIM_ProductPhysicalComponent (System)	Mandatory	Associates the system physical package to its product information.
26.7.11 CIM_SystemPackaging (Component)	Optional	Associates a component system and its physical components.
26.7.12 CIM_SystemPackaging (System)	Mandatory	Associates the top level system and its physical package.

26.7.1 CIM_Container

Associates a PhysicalPackage to its component physical packages (e.g., Drives in a Storage System). This may be subclassed (e.g., PackageInChassis or CardOnCard), but only the Container properties are required.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 251 describes class CIM_Container.

Table 251 - SMI Referenced Properties/Methods for CIM_Container

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	A reference to the higher level physical package.
PartComponent		Mandatory	A reference to a lower level physical package.

26.7.2 CIM_LogicalIdentity

Associates the primary product information to secondary product information. The secondary product information might be the OEM product information.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 252 describes class CIM_LogicalIdentity.

Table 252 - SMI Referenced Properties/Methods for CIM_LogicalIdentity

Properties	Flags	Requirement	Description & Notes
SystemElement		Mandatory	A reference to the primary product information.
SameElement		Mandatory	A reference to a secondary product information.

26.7.3 CIM_PhysicalElementLocation

Associates the physical package of the system to its location.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Support for the Location profile.

Table 253 describes class CIM_PhysicalElementLocation.

Table 253 - SMI Referenced Properties/Methods for CIM_PhysicalElementLocation

Properties	Flags	Requirement	Description & Notes
PhysicalLocation		Mandatory	The reference to the location of the system physical package.
Element		Mandatory	The reference to the system physical package.

26.7.4 CIM_PhysicalPackage (Component)

A physical package for a component of the overall system. There may be multiple instances of a component physical package. For certain component physical packages, this "generic" physical package may be the same as physical packages defined in other component profiles (e.g., this physical package is the same instance as the physical package defined in Disk Drive Lite. It is preferred that component physical packages are modeled in their respective component profiles. However, the component physical package defined in this profile is intended for components (e.g., non-Top Level Systems in the Multiple Computer System Profile) that do not model their physical package.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 254 describes class CIM_PhysicalPackage (Component).

Table 254 - SMI Referenced Properties/Methods for CIM_PhysicalPackage (Component)

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	
Tag		Mandatory	
ElementName		Optional	
Name		Optional	
Manufacturer		Mandatory	
Model		Mandatory	
SerialNumber		Optional	
Version		Optional	
PartNumber		Optional	

26.7.5 CIM_PhysicalPackage (System)

The physical package for the overall system. There shall be only one instance of this class for an autonomous profile (e.g., Array or Fabric).

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 255 describes class CIM_PhysicalPackage (System).

Table 255 - SMI Referenced Properties/Methods for CIM_PhysicalPackage (System)

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	
Tag		Mandatory	
ElementName		Optional	

Table 255 - SMI Referenced Properties/Methods for CIM_PhysicalPackage (System)

Properties	Flags	Requirement	Description & Notes
Name		Optional	
Manufacturer		Mandatory	
Model		Mandatory	
SerialNumber		Optional	
Version		Optional	
PartNumber		Optional	

26.7.6 CIM_Product (Component)

The product information for a physical package that is a component of the system.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 256 describes class CIM_Product (Component).

Table 256 - SMI Referenced Properties/Methods for CIM_Product (Component)

Properties	Flags	Requirement	Description & Notes
Name		Mandatory	
IdentifyingNumber		Mandatory	
Vendor		Mandatory	
Version		Mandatory	
ElementName		Mandatory	

26.7.7 CIM_Product (System)

The product information for the physical package of the system.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 257 describes class CIM_Product (System).

Table 257 - SMI Referenced Properties/Methods for CIM_Product (System)

Properties	Flags	Requirement	Description & Notes
Name		Mandatory	
IdentifyingNumber		Mandatory	
Vendor		Mandatory	

Table 257 - SMI Referenced Properties/Methods for CIM_Product (System)

Properties	Flags	Requirement	Description & Notes
Version		Mandatory	
ElementName		Mandatory	

26.7.8 CIM_ProductParentChild

If more than one product comprises a system, this association should be used to indicate the 'parent' product.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 258 describes class CIM_ProductParentChild.

Table 258 - SMI Referenced Properties/Methods for CIM_ProductParentChild

Properties	Flags	Requirement	Description & Notes
Parent		Mandatory	A reference to the parent (System or Component) product.
Child		Mandatory	A reference to a component product.

26.7.9 CIM_ProductPhysicalComponent (Component)

Associates a component physical package to its product information.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 259 describes class CIM_ProductPhysicalComponent (Component).

Table 259 - SMI Referenced Properties/Methods for CIM_ProductPhysicalComponent (Component)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	A reference to the product information for a component of the system.
PartComponent		Mandatory	A reference to the component physical package.

26.7.10 CIM_ProductPhysicalComponent (System)

Associates the system physical package to its product information.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 260 describes class CIM_ProductPhysicalComponent (System).

Table 260 - SMI Referenced Properties/Methods for CIM_ProductPhysicalComponent (System)

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	The reference to the product information for the system.
PartComponent		Mandatory	The reference to the system physical package.

26.7.11 CIM_SystemPackaging (Component)

Associates a component system and its physical components. The ComputerSystemPackage subclass should be used if the referenced system is subclassed as ComputerSystem.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 261 describes class CIM_SystemPackaging (Component).

Table 261 - SMI Referenced Properties/Methods for CIM_SystemPackaging (Component)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	The component system that has a physical package.
Antecedent		Mandatory	The reference to the PhysicalPackage of the component system.

26.7.12 CIM_SystemPackaging (System)

Associates the top level system and its physical package. The ComputerSystemPackage subclass should be used if the referenced system is subclassed as ComputerSystem.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 262 describes class CIM_SystemPackaging (System).

Table 262 - SMI Referenced Properties/Methods for CIM_SystemPackaging (System)

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	The system that has a physical package.
Antecedent		Mandatory	The reference to the PhysicalPackage of the top level system.

STABLE

STABLE

27 Power Supply Profile

27.1 Synopsis

Profile Name: Power Supply (Component Profile)

Version: 1.0.1

Organization: SNIA

Specializes: DMTF Power Supply Profile

Central Class: PowerSupply

Scoping Class: ComputerSystem

Related Profiles: Table 263 describes the related profiles for Power Supply.

Table 263 - Related Profiles for Power Supply

Profile Name	Organization	Version	Requirement	Description
Physical Asset	DMTF	1.0.1	Optional	See DSP1011, version 1.0.2
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

27.2 Description

The SNIA Power Supply Profile specializes the DMTF Power Supply Profile by adding indications. No other changes are made to the DMTF profile.

27.3 Implementation

See DSP1015: the DMTF Power Supply Profile.

27.3.1 Health and Fault Management Consideration

Not defined in this standard

27.3.2 Cascading Considerations

Not defined in this standard

27.4 Methods

See DSP1015: the DMTF Power Supply Profile.

27.5 Use Cases

See DSP1015: the DMTF Power Supply Profile.

27.6 CIM Elements

Table 264 describes the CIM elements for Power Supply.

Table 264 - CIM Elements for Power Supply

Element Name	Requirement	Description
27.6.1 CIM_ElementCapabilities	Conditional	Conditional requirement: Support for EnabledLogicalElementCapabilities.
27.6.2 CIM_EnabledLogicalElementCapabilities	Optional	
27.6.3 CIM_IsSpare	Optional	
27.6.4 CIM_MemberOfCollection	Conditional	Conditional requirement: Support for Power Supply redundancy.
27.6.5 CIM_OwningCollectionElement	Conditional	Conditional requirement: Support for Power Supply redundancy.
27.6.6 CIM_PowerSupply	Mandatory	
27.6.7 CIM_RedundancySet	Optional	
27.6.8 CIM_SuppliesPower	Optional	
27.6.9 CIM_SystemDevice	Mandatory	
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_PowerSupply	Mandatory	Creation of a PowerSupply instance.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_PowerSupply	Mandatory	Deletion of a PowerSupply instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_PowerSupply AND SourceInstance.CIM_PowerSupply::OperationalStatus <> PreviousInstance.CIM_PowerSupply::OperationalStatus	Mandatory	CQL -Change of Operational Status of a PowerSupply instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_PowerSupply AND SourceInstance.CIM_PowerSupply::EnabledState <> PreviousInstance.CIM_PowerSupply::EnabledState	Mandatory	CQL -Change of EnabledState of a PowerSupply instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_RedundancySet AND SourceInstance.CIM_RedundancySet::RedundancyStatus <> PreviousInstance.CIM_RedundancySet::RedundancyStatus	Conditional	Conditional requirement: Support for Power Supply redundancy. CQL -Change of redundancy status.

27.6.1 CIM_ElementCapabilities

CIM_ElementCapabilities is used to associate CIM_PowerSupply with CIM_EnabledLogicalElementCapabilities that describes the capabilities of CIM_PowerSupply.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for CIM_EnabledLogicalElementCapabilities.

Table 265 describes class CIM_ElementCapabilities.

Table 265 - SMI Referenced Properties/Methods for CIM_ElementCapabilities

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to CIM_PowerSupply.
Capabilities		Mandatory	Reference to CIM_EnabledLogicalElementCapabilities.

27.6.2 CIM_EnabledLogicalElementCapabilities

CIM_EnabledLogicalElementCapabilities represents the capabilities of the power supply.

Requirement: Optional

Table 266 describes class CIM_EnabledLogicalElementCapabilities.

Table 266 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
RequestedStatesSupported		Mandatory	Array that contains the supported requested states for the instance of CIM_PowerSupply. Shall include 2 (Enabled), 3 (Disabled), 6 (Offline), or 11 (Reset).
ElementNameEditSupported		Mandatory	
MaxElementNameLen		Conditional	Conditional requirement: Support for Element Name editing. Conditional on Support for Element Name editing.
ElementName		Mandatory	User-friendly name.

27.6.3 CIM_IsSpare

CIM_IsSpare is used to associate CIM_PowerSupply with CIM_RedundancySet that the CIM_PowerSupply is a member of and where CIM_PowerSupply represents a spare power supply.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 267 describes class CIM_IsSpare.

Table 267 - SMI Referenced Properties/Methods for CIM_IsSpare

Properties	Flags	Requirement	Description & Notes
SpareStatus		Mandatory	Shall be 0 (Unknown), 1 (Cold Standby), or 2 (Hot Standby).
FailoverSupported		Mandatory	Shall be are 2 (Automatic), 3 (Manual) or 4 (Both Manual and Automatic).
Antecedent		Mandatory	The RedundancySet.
Dependent		Mandatory	PowerSupply.

27.6.4 CIM_MemberOfCollection

CIM_MemberOfCollection is used to associate CIM_PowerSupply with CIM_RedundancySet that the CIM_PowerSupply is a member of.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for Power Supply redundancy.

Table 268 describes class CIM_MemberOfCollection.

Table 268 - SMI Referenced Properties/Methods for CIM_MemberOfCollection

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	Reference to RedundancySet
Member		Mandatory	Reference to PowerSupply

27.6.5 CIM_OwningCollectionElement

CIM_OwningCollectionElement is used to associate CIM_RedundancySet with CIM_ComputerSystem that the CIM_RedundancySet is a member of. The instance of CIM_OwningCollectionElement is conditional on having instantiation of the CIM_RedundancySet class.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for Power Supply redundancy.

Table 269 describes class CIM_OwningCollectionElement.

Table 269 - SMI Referenced Properties/Methods for CIM_OwningCollectionElement

Properties	Flags	Requirement	Description & Notes
OwningElement		Mandatory	Reference to ComputerSystem
OwnedElement		Mandatory	Reference to RedundancySet

27.6.6 CIM_PowerSupply

CIM_PowerSupply is used to represent the power supply.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 270 describes class CIM_PowerSupply.

Table 270 - SMI Referenced Properties/Methods for CIM_PowerSupply

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.
DeviceID		Mandatory	Key.
TotalOutputPower		Mandatory	Shall match 0 when the power supply's total output power is unknown.
ElementName		Mandatory	
OperationalStatus		Mandatory	
HealthState		Mandatory	
EnabledState		Mandatory	Shall be 2 (Enabled), 3 (Disabled), 5 (Not Applicable) or 6 (Enabled but Offline).
RequestedState		Mandatory	Shall be 2 (Enabled), 3 (Disabled), 5 (No Change), 6 (Offline), 11 (Reset) or 12 (Not Applicable).
RequestStateChange()		Conditional	Conditional requirement: updating requested states. The implementation shall support this method, but the method may always return 'Not Supported.'

27.6.7 CIM_RedundancySet

CIM_RedundancySet is used to represent the aggregation of redundant power supplies.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 271 describes class CIM_RedundancySet.

Table 271 - SMI Referenced Properties/Methods for CIM_RedundancySet

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	shall be formatted as a free formed string of variable length (pattern ".+").
RedundancyStatus		Mandatory	
TypeOfSet		Mandatory	Shall be 2 (N+1), 3 (Load Balanced), 4 (Sparing) or 5 (Limited Sparing).
MinNumberNeeded		Mandatory	shall match 0 when the minimum number of power supplies needed for the redundancy is unknown.

27.6.8 CIM_SuppliesPower

CIM_SuppliesPower is used to associate CIM_PowerSupply with CIM_ManagedSystemElement that the power supply represented by the CIM_PowerSupply instance supplies power to.

Created By: Static

Modified By: Static
 Deleted By: Static
 Requirement: Optional

Table 272 describes class CIM_SuppliesPower.

Table 272 - SMI Referenced Properties/Methods for CIM_SuppliesPower

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	
Dependent		Mandatory	Shall reference the instance of the subclass of CIM_ManagedSystemElement representing element receiving the power.

27.6.9 CIM_SystemDevice

CIM_SystemDevice is used to associate CIM_PowerSupply with CIM_ComputerSystem that the CIM_PowerSupply is a member of.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 273 describes class CIM_SystemDevice.

Table 273 - SMI Referenced Properties/Methods for CIM_SystemDevice

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	
PartComponent		Mandatory	

STABLE

STABLE
28 Fan Profile**28.1 Synopsis****Profile Name:** Fan (Component Profile)**Version:** 1.5.0**Organization:** SNIA**Specializes:** DMTF Fan Profile**Central Class:** Fan**Scoping Class:** ComputerSystem**Related Profiles:** Table 274 describes the related profiles for Fan.**Table 274 - Related Profiles for Fan**

Profile Name	Organization	Version	Requirement	Description
Physical Asset	DMTF	1.0.2	Optional	See DSP1011, version 1.0.2
Sensors	SNIA	1.5.0	Optional	Experimental.
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

The SNIA Fan Profile specializes DSP1013: the DMTF Fan Profile by adding indications.

28.2 Description

The SNIA Fan Profile specializes the DMTF Fan Profile by adding indications. Other changes are made to the DMTF Profile, include making a couple of classes conditional and elaborating on the definition of RedundancyStatus.

28.3 Implementation

See DSP1013: the DMTF Fan Profile.

28.3.1 Health and Fault Management Consideration

Not defined in this standard

28.3.2 Cascading Considerations

Not defined in this standard

28.4 Methods

See DSP1013: the DMTF Fan Profile.

28.5 Use Cases

See DSP1013: the DMTF Fan Profile.

28.6 CIM Elements

Table 275 describes the CIM elements for Fan.

Table 275 - CIM Elements for Fan

Element Name	Requirement	Description
28.6.1 CIM_AssociatedCooling	Optional	
28.6.2 CIM_AssociatedSensor	Optional	
28.6.3 CIM_ElementCapabilities	Conditional	Conditional requirement: Support for EnabledLogicalElementCapabilities.
28.6.4 CIM_EnabledLogicalElementCapabilities	Optional	
28.6.5 CIM_Fan	Mandatory	
28.6.6 CIM_HostedCollection	Optional	
28.6.7 CIM_IsSpare	Optional	
28.6.8 CIM_MemberOfCollection	Conditional	Conditional requirement: Support for Fan redundancy.
28.6.9 CIM_NumericSensor	Optional	
28.6.10 CIM_OwningCollectionElement	Conditional	Conditional requirement: Support for Fan redundancy.
28.6.11 CIM_RedundancySet (Fan Redundancy)	Optional	
28.6.12 CIM_Sensor	Optional	
28.6.13 CIM_SystemDevice	Mandatory	
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_Fan	Mandatory	Creation of a Fan instance.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_Fan	Mandatory	CQL -Deletion of a Fan instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_Fan AND SourceInstance.CIM_Fan::OperationalStatus <> PreviousInstance.CIM_Fan::OperationalStatus	Mandatory	CQL -Change of Operational Status of a Fan instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_Fan AND SourceInstance.CIM_Fan::EnabledState <> PreviousInstance.CIM_Fan::EnabledState	Mandatory	CQL -Change of EnabledState of a Fan instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_RedundancySet AND SourceInstance.CIM_RedundancySet::RedundancyStatus <> PreviousInstance.CIM_RedundancySet::RedundancyStatus	Conditional	Conditional requirement: Support for Fan redundancy. CQL -Change of redundancy status.

28.6.1 CIM_AssociatedCooling

CIM_AssociatedCooling associates CIM_Fan with a subclass of CIM_ManagedSystemElement.

Requirement: Optional

Table 276 describes class CIM_AssociatedCooling.

Table 276 - SMI Referenced Properties/Methods for CIM_AssociatedCooling

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to CIM_Fan.
Dependent		Mandatory	Shall reference an instance of a subclass of CIM_ManagedSystemElement for which the fan is providing cooling.

28.6.2 CIM_AssociatedSensor

This is described in the DMTF profile, but is missing from the CIM Elements table.

Requirement: Optional

Table 277 describes class CIM_AssociatedSensor.

Table 277 - SMI Referenced Properties/Methods for CIM_AssociatedSensor

Properties	Flags	Requirement	Description & Notes
Dependent		Mandatory	Reference to CIM_Fan.
Antecedent		Mandatory	Reference to Sensor or NumericSensor.

28.6.3 CIM_ElementCapabilities

SNIA makes this conditional on the existence of CIM_EnabledLogicalElementCapabilities. The class definition specializes the CIM_ElementCapabilities definition in the Fan profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for CIM_EnabledLogicalElementCapabilities.

Table 278 describes class CIM_ElementCapabilities.

Table 278 - SMI Referenced Properties/Methods for CIM_ElementCapabilities

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to CIM_Fan.
Capabilities		Mandatory	Reference to CIM_EnabledLogicalElementCapabilities.

28.6.4 CIM_EnabledLogicalElementCapabilities

CIM_EnabledLogicalElementCapabilities represents the capabilities of the Fan.

Requirement: Optional

Table 279 describes class CIM_EnabledLogicalElementCapabilities.

Table 279 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
RequestedStatesSupported	N	Mandatory	Array that contains the supported requested states for the instance of CIM_Fan. Shall be an empty array or contain any combination of: 2 (Enabled), 3 (Disabled), or 11 (Reset).
ElementNameEditSupported		Mandatory	property shall have a value of TRUE when the implementation supports client modification of the ElementName property of the associated instance of CIM_Fan.
MaxElementNameLen		Conditional	Conditional requirement: The ElementNameEditSupported property has a value of TRUE. Conditional on Support for Element Name editing.

28.6.5 CIM_Fan

CIM_Fan is used to represent the fan.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 280 describes class CIM_Fan.

Table 280 - SMI Referenced Properties/Methods for CIM_Fan

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.
DeviceID		Mandatory	Key.
OperationalStatus		Mandatory	
HealthState		Mandatory	
VariableSpeed		Mandatory	
DesiredSpeed		Conditional	Conditional requirement: Support for the SetSpeed method (VariableSpeed=\TRUE\)andEnabledStateisNOT3(Disabled).'
ActiveCooling		Mandatory	Shall have the value TRUE.
EnabledState		Mandatory	This property shall match the values 0 (Unknown), 2(Enabled), 3(Disabled) or 5(Not Applicable).
RequestedState		Mandatory	This property shall have a value of 2 (Enabled), 3 (Disabled) , 5 (No Change), 11 (Reset) or 12 (Not Applicable).
ElementName		Mandatory	This property shall be modifiable when the ElementNameEditSupported property of the associated CIM_EnabledLogicalElementCapabilities instance has a value of TRUE.

Table 280 - SMI Referenced Properties/Methods for CIM_Fan

Properties	Flags	Requirement	Description & Notes
SetSpeed()		Conditional	Conditional requirement: Support for the SetSpeed method (VariableSpeed=\TRUE\)andEnabledStateisNOT3(Disabled).'
RequestStateChange()		Conditional	Conditional requirement: The RequestedStatesSupported property has a value of 2 (Enabled), 3 (Disabled) or 11 (Reset)in the CIM_EnabledLogicalElementCapabilities for the Fan..

28.6.6 CIM_HostedCollection

HostedCollection defines a SystemSpecificCollection in the context of a scoping System.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 281 describes class CIM_HostedCollection.

Table 281 - SMI Referenced Properties/Methods for CIM_HostedCollection

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Key: Shall reference a CIM_ComputerSystem instance of which a CIM_RedundancySet instance is a member.
Dependent		Mandatory	Key: Shall reference a CIM_RedundancySet instance.

28.6.7 CIM_IsSpare

CIM_IsSpare is used to associate CIM_Fan with CIM_RedundancySet that the CIM_Fan is a member of and where CIM_Fan represents a spare Fan.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 282 describes class CIM_IsSpare.

Table 282 - SMI Referenced Properties/Methods for CIM_IsSpare

Properties	Flags	Requirement	Description & Notes
SpareStatus		Mandatory	Shall be 0 (Unknown) or 3 (Cold Standby).
FailoverSupported		Mandatory	Shall be 2 (Automatic), 3 (Manual) or 4 (Both Manual and Automatic).
Antecedent		Mandatory	Key: Shall reference a CIM_RedundancySet instance of which a CIM_Fan instance is a member and where the CIM_Fan instance is a Spare Fan.
Dependent		Mandatory	Key: Shall reference the Spare Fan.

28.6.8 CIM_MemberOfCollection

CIM_MemberOfCollection associates CIM_Fan with the CIM_RedundancySet instance of which CIM_Fan is a member.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for Fan redundancy.

Table 283 describes class CIM_MemberOfCollection.

Table 283 - SMI Referenced Properties/Methods for CIM_MemberOfCollection

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	Key: Shall reference a CIM_RedundancySet instance of which a CIM_Fan instance is a member.
Member		Mandatory	Key: Shall reference a CIM_Fan instance.

28.6.9 CIM_NumericSensor

The CIM_NumericSensor class is defined by the Sensors Profile. The requirements denoted here are in addition to those mandated by the Sensors Profile.

Requirement: Optional

Table 284 describes class CIM_NumericSensor.

Table 284 - SMI Referenced Properties/Methods for CIM_NumericSensor

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.
DeviceID		Mandatory	Key.
SensorType		Mandatory	Shall be set to 5 (Tachometer).
BaseUnits		Mandatory	Shall be 19 (RPM).
RateUnits		Mandatory	Shall be 0 (None).

28.6.10 CIM_OwningCollectionElement

SNIA makes this conditional on the existence of CIM_RedundancySet. The class definition specializes the CIM_OwningCollectionElement definition in the Fan profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for Fan redundancy.

Table 285 describes class CIM_OwningCollectionElement.

Table 285 - SMI Referenced Properties/Methods for CIM_OwningCollectionElement

Properties	Flags	Requirement	Description & Notes
OwnedElement		Mandatory	Key: Shall reference a CIM_RedundancySet instance.
OwningElement		Mandatory	Key: Shall reference a CIM_ComputerSystem instance of which a CIM_RedundancySet instance is a member.

28.6.11 CIM_RedundancySet (Fan Redundancy)

SNIA specializes this to further define the RedundancyStatus property. The class definition specializes the CIM_RedundancySet definition in the Fan profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 286 describes class CIM_RedundancySet (Fan Redundancy).

Table 286 - SMI Referenced Properties/Methods for CIM_RedundancySet (Fan Redundancy)

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	Key.
ElementName		Mandatory	shall be formatted as a free formed string of variable length (pattern ".*").
RedundancyStatus		Mandatory	Shall be 2 (Fully Redundant), 3 (Degraded Redundancy), 4 (Redundancy Lost) or 5 (Overall Failure).
TypeOfSet		Mandatory	Shall be 2 (N+1), 3 (Load Balanced), 4 (Sparing) or 5 (Limited Sparing).
MinNumberNeeded		Mandatory	Shall be 0 when the minimum number of fans needed for the redundancy is unknown.
Failover()		Optional	

28.6.12 CIM_Sensor

The CIM_Sensor class is defined by the Sensors Profile. The requirements denoted here are in addition to those mandated by the Sensors Profile.

Requirement: Optional

Table 287 describes class CIM_Sensor.

Table 287 - SMI Referenced Properties/Methods for CIM_Sensor

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.

Table 287 - SMI Referenced Properties/Methods for CIM_Sensor

Properties	Flags	Requirement	Description & Notes
DeviceID		Mandatory	Key.
SensorType		Mandatory	Shall be set to 5 (Tachometer).

28.6.13 CIM_SystemDevice

CIM_SystemDevice associates CIM_Fan with the CIM_ComputerSystem instance of which CIM_Fan is a member.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 288 describes class CIM_SystemDevice.

Table 288 - SMI Referenced Properties/Methods for CIM_SystemDevice

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Key: Shall reference a CIM_ComputerSystem instance of which a CIM_Fan instance is a member.
PartComponent		Mandatory	Key: Shall reference a CIM_Fan instance.

STABLE

EXPERIMENTAL
29 Sensors Profile**29.1 Synopsis****Profile Name:** Sensors (Component Profile)**Version:** 1.5.0**Organization:** SNIA**Specializes:** DMTF Sensors Profile**Central Class:** Sensor**Scoping Class:** ComputerSystem**Related Profiles:** Table 289 describes the related profiles for Sensors.**Table 289 - Related Profiles for Fan**

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

The SNIA Sensors Profile specializes DSP1009: the DMTF Sensors Profile by adding indications.

29.2 Description

The SNIA Sensors Profile specializes the DMTF Sensors Profile by adding indications. No other changes are made to the DMTF Profile.

29.3 Implementation

See DSP1009: the DMTF Sensors Profile.

29.3.1 Health and Fault Management Consideration

Not defined in this standard

29.3.2 Cascading Considerations

Not defined in this standard

29.4 Methods

See DSP1009: the DMTF Sensors Profile.

29.5 Use Cases

See DSP1009: the DMTF Sensors Profile.

29.6 CIM Elements

Table 290 describes the CIM elements for Sensors.

Table 290 - CIM Elements for Sensors

Element Name	Requirement	Description
29.6.1 CIM_AssociatedSensor	Optional	This is used to associate the instance of Sensor (or NumericSensor) with the instance of a subclass of ManagedElement.
29.6.2 CIM_ElementCapabilities	Optional	This is used to associate Sensor (or NumericSensor) with an instance of EnabledLogicalElementCapabilities that describes the capabilities of Sensor.
29.6.3 CIM_EnabledLogicalElementCapabilities	Optional	This is used to represent the capabilities of the sensor as it applies to the properties of Sensor.
29.6.4 CIM_NumericSensor	Optional	
29.6.5 CIM_Sensor	Mandatory	The implementation shall instantiate an instance of Sensor, but this could be the subclass NumericSensor.
29.6.6 CIM_SystemDevice	Mandatory	SystemDevice is used to associate the instance of Sensor (or NumericSensor) with the instance of ComputerSystem of which the Sensor instance is a member.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_Sensor	Mandatory	Creation of a Sensor instance.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_Sensor	Mandatory	Deletion of a Sensor instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_Sensor AND SourceInstance.CIM_Sensor::OperationalStatus <> PreviousInstance.CIM_Sensor::OperationalStatus	Mandatory	CQL -Change of Operational Status of a Sensor instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_Sensor AND SourceInstance.CIM_Sensor::EnabledState <> PreviousInstance.CIM_Sensor::EnabledState	Mandatory	CQL -Change of EnabledState of a Sensor instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_Sensor AND SourceInstance.CIM_Sensor::CurrentState <> PreviousInstance.CIM_Sensor::CurrentState	Mandatory	CQL -Change of Current State of a Sensor instance.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_NumericSensor AND SourceInstance.CIM_Sensor::CurrentReading <> PreviousInstance.CIM_Sensor::CurrentReading	Mandatory	CQL -Change of Current Reading of a Sensor instance.

29.6.1 CIM_AssociatedSensor

CIM_AssociatedSensor associates CIM_Sensor or CIM_NumericSensor with a subclass of CIM_ManagedSystemElement.

Requirement: Optional

Table 291 describes class CIM_AssociatedSensor.

Table 291 - SMI Referenced Properties/Methods for CIM_AssociatedSensor

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Shall be a reference to a specific instance of CIM_Sensor or CIM_NumericSensor.
Dependent		Mandatory	Shall reference an instance of a subclass of CIM_ManagedElement for which the sensor is monitoring.

29.6.2 CIM_ElementCapabilities

CIM_ElementCapabilities is used to associate CIM_Sensor with the CIM_EnabledLogicalElementCapabilities instance that describes the capabilities of the Sensor.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 292 describes class CIM_ElementCapabilities.

Table 292 - SMI Referenced Properties/Methods for CIM_ElementCapabilities

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to Sensor
Capabilities		Mandatory	Reference to EnabledLogicalElementCapabilities

29.6.3 CIM_EnabledLogicalElementCapabilities

CIM_EnabledLogicalElementCapabilities represents the capabilities of the Sensor.

Requirement: Optional

Table 293 describes class CIM_EnabledLogicalElementCapabilities.

Table 293 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	Key.
RequestedStatesSupported		Mandatory	This property is an array that contains the supported requested states for the instance of CIM_Sensor or CIM_NumericSensor. This property shall be the super set of the values to be used as the RequestedState parameter in the RequestStateChange() method. The valid values of the property shall be an empty array or contain any combination of the following values: 2 (Enabled), 3 (Disabled), or 11 (Reset).

Table 293 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities

Properties	Flags	Requirement	Description & Notes
ElementNameEditSupported		Mandatory	This property shall have a value of TRUE when the implementation supports client modification of the ElementName property of the associated CIM_Sensor or CIM_NumericSensor instance.
MaxElementNameLen		Conditional	Conditional requirement: This property shall be implemented when the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE.

29.6.4 CIM_NumericSensor

CIM_NumericSensor is used to represent an analog sensor. The CIM_NumericSensor class is mandatory when the CIM_Sensor class is not implemented. The class definition specializes the CIM_NumericSensor definition in the Sensors profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 294 describes class CIM_NumericSensor.

Table 294 - SMI Referenced Properties/Methods for CIM_NumericSensor

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.
DeviceID		Mandatory	Key.
BaseUnits		Mandatory	None.
UnitModifier		Mandatory	None.
RateUnits		Mandatory	None.
CurrentReading		Mandatory	None.
LowerThresholdNonCritical		Conditional	Conditional requirement: This property shall be mandatory when the CIM_NumericSensor.SupportedThresholds array contains a value of 0 (LowerThresholdNonCritical). This property shall be settable only if the CIM_NumericSensor.SettableThresholds array contains a value of 0 (LowerThresholdNonCritical).
UpperThresholdNonCritical		Conditional	Conditional requirement: This property shall be mandatory when the CIM_NumericSensor.SupportedThresholds array contains a value of 1 (UpperThresholdNonCritical). This property shall be settable only if the CIM_NumericSensor.SettableThresholds array contains a value of 1 (UpperThresholdNonCritical).
LowerThresholdCritical		Conditional	Conditional requirement: This property shall be mandatory when the CIM_NumericSensor.SupportedThresholds array contains a value of 2 (LowerThresholdCritical). This property shall be settable only if the CIM_NumericSensor.SettableThresholds array contains a value of 2 (LowerThresholdCritical).

Table 294 - SMI Referenced Properties/Methods for CIM_NumericSensor

Properties	Flags	Requirement	Description & Notes
UpperThresholdCritical		Conditional	Conditional requirement: This property shall be mandatory when the CIM_NumericSensor.SupportedThresholds array contains a value of 3 (UpperThresholdCritical). This property shall be settable only if the CIM_NumericSensor.SettableThresholds array contains a value of 3 (UpperThresholdCritical).
LowerThresholdFatal		Conditional	Conditional requirement: This property shall be mandatory when the CIM_NumericSensor.SupportedThresholds array contains a value of 4 (LowerThresholdFatal). This property shall be settable only if the CIM_NumericSensor.SettableThresholds array contains a value of 4 (LowerThresholdFatal).
UpperThresholdFatal		Conditional	Conditional requirement: This property shall be mandatory when the CIM_NumericSensor.SupportedThresholds array contains a value of 5 (UpperThresholdFatal). This property shall be settable only if the CIM_NumericSensor.SettableThresholds array contains a value of 5 (UpperThresholdFatal).
SupportedThresholds	N	Mandatory	This property is an array that contains the list of the implemented thresholds: 0 (LowerThresholdNonCritical), 1 (UpperThresholdNonCritical), 2 (LowerThresholdCritical), 3 (UpperThresholdCritical), 4 (LowerThresholdFatal), and 5 (UpperThresholdFatal). When the implementation does not support any of these threshold properties, the property shall be an empty array.
SettableThresholds	N	Mandatory	This property is an array that contains the list of the settable implemented thresholds: 0 (LowerThresholdNonCritical), 1 (UpperThresholdNonCritical), 2 (LowerThresholdCritical), 3 (UpperThresholdCritical), 4 (LowerThresholdFatal), and 5 (UpperThresholdFatal). The array shall contain the subset of values in the CIM_NumericSensor.SupportedThresholds array. When the implementation does not support any of the settable threshold properties, the property shall be an empty array.
SensorType		Mandatory	None.
PossibleStates		Mandatory	See section 7.3 of the DMTF Sensors Profile version 1.0.2.
CurrentState		Mandatory	The CIM_NumericSensor.CurrentState property shall have a value of one of the elements in the CIM_NumericSensor.PossibleStates array.
ElementName		Mandatory	The CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property shall have a value of TRUE when the implementation supports client modification of the ElementName property of the associated CIM_NumericSensor instance.
OtherSensorTypeDescription		Conditional	Conditional requirement: The OtherSensorTypeDescription property shall be mandatory when the SensorType property is set to a value of 1 (Other). The OtherSensorTypeDescription property shall be formatted as a free-formed string of variable length (pattern \.*\').
EnabledState		Mandatory	This property may have the values 2 (Enabled), 3 (Disabled) or 5 (Not Applicable).
RequestedState		Mandatory	The RequestedState property shall have a value of 12 (Not Applicable), a value of 5 (No Change), or a value that is contained in the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property array of the associated CIM_EnabledLogicalElementCapabilities instance (i.e., 2 (Enabled), 3 (Disabled) or 11(Reset)).
OperationalStatus		Mandatory	None.
HealthState		Mandatory	None.

Table 294 - SMI Referenced Properties/Methods for CIM_NumericSensor

Properties	Flags	Requirement	Description & Notes
RequestStateChange()		Conditional	Conditional requirement: When a CIM_EnabledLogicalElementCapabilities instance is associated with the Central Instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property is a non-empty array, sensor state management shall be supported.
RestoreDefaultThresholds()		Conditional	Conditional requirement: The CIM_NumericSensor.RestoreDefaultThresholds() method shall be implemented and shall not return a value of 1 (Unsupported) when the CIM_NumericSensor.SettableThresholds property is a non-empty array.

29.6.5 CIM_Sensor

CIM_Sensor is used to represent a discrete sensor. Either the CIM_Sensor class (or its subclass) CIM_NumericSensor class is mandatory. The class definition specializes the CIM_Sensor definition in the Sensors profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 295 describes class CIM_Sensor.

Table 295 - SMI Referenced Properties/Methods for CIM_Sensor

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.
DeviceID		Mandatory	Key.
SensorType		Mandatory	none.
PossibleStates		Mandatory	See section 7.2 of the DMTF Sensors Profile version 1.0.2.
CurrentState		Mandatory	The CIM_Sensor.CurrentState property shall have a value of one of the elements in the CIM_Sensor.PossibleStates array.
ElementName		Mandatory	The CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property shall have a value of TRUE when the implementation supports client modification of the ElementName property of the associated CIM_Sensor.
OtherSensorTypeDescription		Conditional	Conditional requirement: The OtherSensorTypeDescription property shall be mandatory when the SensorType property is set to a value of 1 (Other).The OtherSensorTypeDescription property shall be formatted as a free-formed string of variable length (pattern \.*\').'
EnabledState		Mandatory	This property may have the values 2 (Enabled), 3 (Disabled) or 5 (Not Applicable).

Table 295 - SMI Referenced Properties/Methods for CIM_Sensor

Properties	Flags	Requirement	Description & Notes
RequestedState		Mandatory	The RequestedState property shall have a value of 12 (Not Applicable), a value of 5 (No Change), or a value that is contained in the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property array of the associated CIM_EnabledLogicalElementCapabilities instance (i.e., 2 (Enabled), 3 (Disabled) or 11(Reset)).
OperationalStatus		Mandatory	none.
HealthState		Mandatory	none.
RequestStateChange()		Conditional	Conditional requirement: When a CIM_EnabledLogicalElementCapabilities instance is associated with the Central Instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property is a non-empty array, sensor state management shall be supported.

29.6.6 CIM_SystemDevice

CIM_SystemDevice is used to associate CIM_Sensor with CIM_ComputerSystem that the CIM_Sensor is a member of.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 296 describes class CIM_SystemDevice.

Table 296 - SMI Referenced Properties/Methods for CIM_SystemDevice

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Key: shall be a reference to the CIM_ComputerSystem instance of which the current CIM_Sensor (or CIM_NumericSensor) instance is a member.
PartComponent		Mandatory	Key: shall be a reference to the current CIM_Sensor (or CIM_NumericSensor) instance.

EXPERIMENTAL

Sensors Profile

EXPERIMENTAL

30 Base Server Profile

30.1 Synopsis

Profile Name: Base Server (Autonomous Profile)

Version: 1.6.0

Organization: SNIA

Specializes: DMTF Base Server 1.0.1 to reference SNIA specialized referenced profiles.

Central Class: CIM_ComputerSystem

Scoping Class: CIM_ComputerSystem

Related Profiles: Table 297 describes the related profiles for Base Server.

Table 297 - Related Profiles for Base Server

Profile Name	Organization	Version	Requirement	Description
Computer System	DMTF	1.0.3	Optional	See DSP1052, version 1.0.3
Fan	SNIA	1.5.0	Optional	Experimental.
Physical Asset	DMTF	1.0.2	Mandatory	See DSP1011 version 1.0.2
Power Supply	SNIA	1.0.1	Optional	Experimental.
Record Log	DMTF	1.0.1	Optional	See DSP1052 version 1.0.3, section 5
Sensors	SNIA	1.5.0	Optional	Experimental.
Software Inventory	SNIA	1.0.0	Optional	Experimental.
Software Update	DMTF	1.0.0	Optional	See DSP1025. version 1.0.0, section 5
Storage HBA	SNIA	1.6.1	Optional	Experimental.
Host Discovered Resources	SNIA	1.6.0	Optional	Experimental.
Disk Partition	SNIA	1.6.0	Optional	Experimental.
SCSI Multipath Management	SNIA	1.6.0	Optional	Experimental.
Host Hardware RAID Controller	SNIA	1.5.0	Optional	
Storage Enclosure	SNIA	1.3.0	Optional	Experimental.
Launch In Context	DMTF	1.0.0	Optional	Experimental. See DSP1102 version 1.0.0
Host Filesystem	SNIA	1.6.0	Optional	Experimental.

The Base Server Profile models a customer server or storage system.

30.2 Description

The SNIA Base Server Profile models a customer server or storage system containing storage elements. This profile may be used to scope one or more HBAs (or other storage elements).

This profile may represent either a physical system or a virtual system.

30.3 Implementation

See DSP1004, DMTF Base Server Profile for details on the model.

In a storage context, there are several related deployment options.

30.3.1 HBA Instrumentation

If an HBA vendor wishes to create HBA instrumentation that can be used with CIM instrumentation from a server vendor, they would implement the component Storage HBA Profile and work with the server vendor(s) to assure it integrates effectively with their autonomous server profile. If an HBA vendor wishes to deliver a free-standing implementation that does not rely on server-vendor software, they could implement this profile along with the Storage HBA Profile. Note that the HBA vendor could support both approaches and let a customer or installation script determine which is most appropriate.

30.3.2 Host Hardware RAID Instrumentation

Host Hardware RAID vendors have the same deployment options as HBA vendors (see 30.3.1)

30.3.3 Storage Enclosure Instrumentation

In configurations where the Storage Enclosure Profile is not used with a single autonomous profile, the Base Server may be used as the referencing profile for the Storage Enclosure and other component profiles.

30.3.4 Health and Fault Management Consideration

Not defined in this standard

30.3.5 Cascading Considerations

Not defined in this standard

30.4 Methods

See DSP1004, DMTF Base Server Profile.

30.5 Use Cases

See DSP1004, DMTF Base Server Profile.

30.6 CIM Elements

Table 298 describes the CIM elements for Base Server.

Table 298 - CIM Elements for Base Server

Element Name	Requirement	Description
30.6.1 CIM_ComputerSystem	Mandatory	The hosting system for the Storage Elements. Associated to RegisteredProfile.
30.6.2 CIM_ComputerSystemPackage	Mandatory	DSP1004 version 1.0.1, section 10.2.
30.6.3 CIM_ElementCapabilities	Optional	See DSP1052 version 1.0.3, section 10.2.

Table 298 - CIM Elements for Base Server

Element Name	Requirement	Description
30.6.4 CIM_EnabledLogicalElementCapabilities	Optional	See DSP1052 version 1.0.3, section 10.3 and DSP1004 version 1.0.1, section 10.2.
30.6.5 CIM_HostedService	Optional	See DSP1052 version 1.0.3, section 10.4.
30.6.6 CIM_PhysicalPackage	Mandatory	DSP1004 version 1.0.1, section 10.4 and DSP1011 version 1.0.2, section 10.16.
30.6.7 CIM_ServiceAffectsElement	Optional	See DSP1052 version 1.0.3, section 10.5.
30.6.8 CIM_TimeService	Optional	Experimental. See DSP1052 version 1.0.3, section 10.6.

30.6.1 CIM_ComputerSystem

The hosting system for the Storage Elements. The class definition specializes the CIM_ComputerSystem definition in the Base Server profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Shall be associated to RegisteredProfile using ElementConformsToProfile association. The RegisteredProfile instance shall have RegisteredName set to 'Base Server', RegisteredOrganization set to 'SNIA', and RegisteredVersion set to '1.6.0'.

Table 299 describes class CIM_ComputerSystem.

Table 299 - SMI Referenced Properties/Methods for CIM_ComputerSystem

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	Key.
Name		Mandatory	Unique identifier for the hosting system.
ElementName		Mandatory	User friendly name.
OtherIdentifyingInfo		Mandatory	
IdentifyingDescriptions		Optional	Required when OtherIdentifyingInfo is 1. See DSP1052 version 1.0.3, section 10.1.
Dedicated		Mandatory	0 (Not Dedicated).
OperationalStatus		Mandatory	See DSP1052 version 1.0.3, section 10.1.
HealthState		Mandatory	See DSP1052 version 1.0.3, section 10.1.
EnabledState		Mandatory	See DSP1052 version 1.0.3, section 10.1 and DSP1004 version 1.0.1, section 10.1.
RequestedState		Mandatory	See DSP1052 version 1.0.3, section 10.1 and DSP1004 version 1.0.1, section 10.1.
NameFormat		Mandatory	
RequestStateChange()		Conditional	Conditional requirement: The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains at least one value. See DSP1052 version 1.0.3, section 10.1.

30.6.2 CIM_ComputerSystemPackage

One or more instances of CIM_ComputerSystemPackage associate the CIM_ComputerSystem instance with the CIM_PhysicalPackage instances in which it resides. The constraints specified in this Table are in addition to those specified in the Physical Asset Profile.

Requirement: Mandatory

Table 300 describes class CIM_ComputerSystemPackage.

Table 300 - SMI Referenced Properties/Methods for CIM_ComputerSystemPackage

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to PhysicalPackage

30.6.3 CIM_ElementCapabilities

CIM_ElementCapabilities associates an instance of CIM_EnabledLogicalElementCapabilities with an instance of CIM_ComputerSystem.

Requirement: Optional

Table 301 describes class CIM_ElementCapabilities.

Table 301 - SMI Referenced Properties/Methods for CIM_ElementCapabilities

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	This property shall be a reference to an instance of CIM_ComputerSystem.
Capabilities		Mandatory	This property shall be a reference to the instance of CIM_EnabledLogicalElementCapabilities.

30.6.4 CIM_EnabledLogicalElementCapabilities

CIM_EnabledLogicalElementCapabilities indicates support for managing the state of the system.

Requirement: Optional

Table 302 describes class CIM_EnabledLogicalElementCapabilities.

Table 302 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	Key.
RequestedStatesSupported		Mandatory	See DSP1052 version 1.0.3, section 10.3 and DSP1004 version 1.0.1, section 10.3.
ElementNameEditSupported		Mandatory	See DSP1052 version 1.0.3, section 10.3.
MaxElementNameLen		Optional	See DSP1052 version 1.0.3, section 10.3.

30.6.5 CIM_HostedService

CIM_HostedService relates the CIM_TimeService to its scoping CIM_ComputerSystem instance.

Requirement: Optional

Table 303 describes class CIM_HostedService.

Table 303 - SMI Referenced Properties/Methods for CIM_HostedService

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	This property shall reference the Central Instance.
Dependent		Mandatory	This property shall reference TimeService.

30.6.6 CIM_PhysicalPackage

One or more instances of CIM_PhysicalPackage represent the physical packaging of the computer system. Other than the existence of at least one instance of CIM_PhysicalPackage, this profile does not specify any constraints for CIM_PhysicalPackage beyond those specified in the Physical Asset Profile.

Requirement: Mandatory

Table 304 describes class CIM_PhysicalPackage.

Table 304 - SMI Referenced Properties/Methods for CIM_PhysicalPackage

Properties	Flags	Requirement	Description & Notes
Tag		Mandatory	Key.
CreationClassName		Mandatory	Key.
PackageType		Mandatory	See DSP1011 version 1.0.2, section 10.16.
Manufacturer		Conditional	Conditional requirement: The CIM_PhysicalAssetCapabilities.FRUInfoSupported has a value of TRUE. See DSP1011 version 1.0.2, section 10.16.
Model		Conditional	Conditional requirement: The CIM_PhysicalAssetCapabilities.FRUInfoSupported has a value of TRUE. See DSP1011 version 1.0.2 section 10.16.
SerialNumber		Conditional	Conditional requirement: The CIM_PhysicalAssetCapabilities.FRUInfoSupported has a value of TRUE. See DSP1011 version 1.0.2, section 10.16.
PartNumber		Conditional	Conditional requirement: The CIM_PhysicalAssetCapabilities.FRUInfoSupported has a value of TRUE. See DSP1011 version 1.0.2, section 10.16.
SKU		Conditional	Conditional requirement: The CIM_PhysicalAssetCapabilities.FRUInfoSupported has a value of TRUE. See DSP1011 version 1.0.2, section 10.16.
VendorCompatibilityStrings		Optional	See DSP1011 version 1.0.2, section 10.16.
CanBeFRUed		Optional	This property should be implemented when the Physical Element can be replaced in the field.
Version		Optional	The property shall be the hardware version.

Table 304 - SMI Referenced Properties/Methods for CIM_PhysicalPackage

Properties	Flags	Requirement	Description & Notes
Name		Optional	
ElementName		Mandatory	This property shall be formatted as a free-form string of variable length.

30.6.7 CIM_ServiceAffectsElement

CIM_ServiceAffectsElement associates the CIM_TimeService instance with the Central Instance.

Requirement: Optional

Table 305 describes class CIM_ServiceAffectsElement.

Table 305 - SMI Referenced Properties/Methods for CIM_ServiceAffectsElement

Properties	Flags	Requirement	Description & Notes
ElementEffects		Mandatory	Matches 5 (Manages).
AffectedElement		Mandatory	This property shall be a reference to the Central Instance.
AffectingElement		Mandatory	This property shall be a reference to an instance of CIM_TimeService.

30.6.8 CIM_TimeService

Experimental. CIM_TimeService manages the current time on the system.

Requirement: Optional

Table 306 describes class CIM_TimeService.

Table 306 - SMI Referenced Properties/Methods for CIM_TimeService

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	Key.
SystemName		Mandatory	Key.
CreationClassName		Mandatory	Key.
Name		Mandatory	Key.
ElementName		Mandatory	See DSP1052 version 1.0.3, section 10.6.
ManageTime()		Mandatory	See DSP1052 version 1.0.3, section 10.6.

EXPERIMENTAL

EXPERIMENTAL

31 Media Access Device Profile

31.1 Synopsis

Profile Name: Media Access Device (Component Profile)

Version: 1.0.0

Organization: SNIA

Central Class: MediaAccessDevice

Scoping Class: ComputerSystem

Related Profiles: Table 307 describes the related profiles for Media Access Device.

Table 307 - Related Profiles for Media Access Device

Profile Name	Organization	Version	Requirement	Description
Software Inventory	SNIA	1.0.0	Mandatory	
Software Update	DMTF	1.0.0	Optional	See DSP1025, version 1.0.0
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

The Media Access Device Profile models media access devices - such as tape and CD drives.

31.2 Description

The Media Access Device Profile models media access devices - such as tape and CD drives.

31.2.1 Location Indicator

The implementation may optionally support a drive location indicator (such as an LED) using CIM_MediaAccessDevice.LocationIndicator. The client may set this to 2 (On) or 3 (Off). If the implementation does not support this feature, LocationIndicator shall have the value 4 (Not Supported).

31.2.2 Media Access Device Online/Offline

The drive may be started or stopped by setting the Starting and Stopping values in OperationalStatus using the RequestStateChange method. Figure 42 shows Media Access Device Class information.

See Table 308.

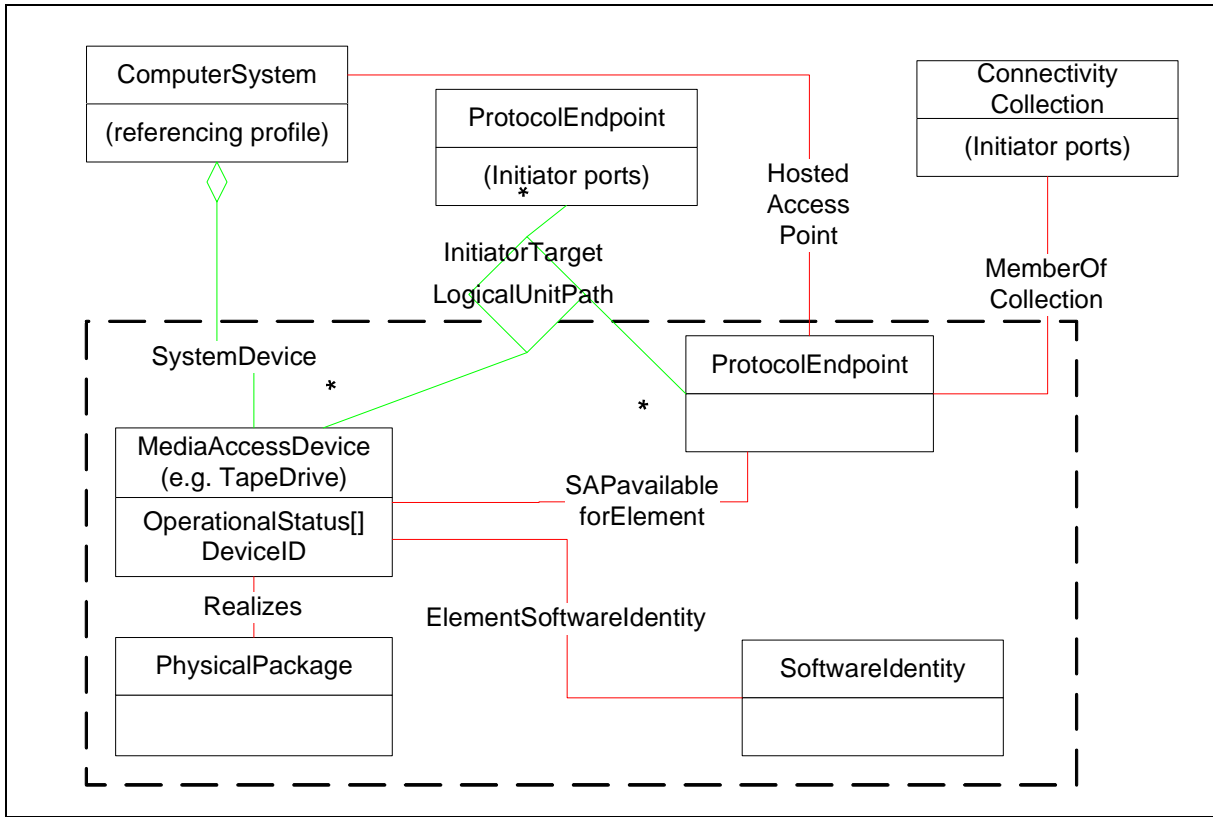


Figure 42 - Media Access Device Class Diagram

31.3 Implementation

31.3.1 Health and Fault Management Consideration

The MediaAccessDevice.OperationalStatus contains the overall status of the disk, summarized in Table 308.

Table 308 - OperationalStatus For MediaAccessDevice

Primary Operational Status	Subsidiary Operational Status	Description
2 "OK"		Media Access Device is enabled.
5 "Predictive Failure"		Media Access Device is functionality nominally but is predicting a failure
6 "Error"		Media Access Device is no longer functioning.
8 "Starting"		Media Access Device is becoming enabled.
9 "Stopping"		Media Access Device is being disabled.
10 "Stopped"		Media Access Device is disabled.

31.3.2 Cascading Considerations

Not defined in this standard.

31.3.3 Hot swap insertion or Removal of Drives

Insertion of a drive shall cause an InstCreation indication for the MediaAccessDevice instance. Similarly, hot-swap removal shall cause an InstDelete indication. ProtocolEndpoint, PhysicalPackage, SoftwareInventory, and related associations will also be created and deleted when a drive is inserted or removed, but no indications shall be produced for these other classes.

31.4 Methods

31.4.1 Request State Change

```
uint32 RequestStateChange(
    [In] uint16 RequestedState,
    [Out] CIM_ConcreteJob REF Job,
    [In] datetime TimeoutPeriod)
```

The allowed state changes are indicated by the RequestedStatesSupported property of EnabledLogicalElementCapabilities. A Job shall be returned if the operation takes longer than the TimeoutPeriod. The Requested State of Offline makes a drives extents unavailable to the dependent volume.

The Job may represent a drive rebuild if the RequestedState of the drive is Offline and a failover shall be complete before the offline operation can finish.

31.5 Use Cases

Not defined in this standard.

31.6 CIM Elements

Table 309 describes the CIM elements for Media Access Device.

Table 309 - CIM Elements for Media Access Device

Element Name	Requirement	Description
31.6.1 CIM_EnabledLogicalElementCapabilities	Mandatory	
31.6.2 CIM_HostedAccessPoint	Optional	ComputerSystem to storage ProtocolEndpoint.
31.6.3 CIM_MediaAccessDevice	Mandatory	Represents a tape or optical drive.
31.6.4 CIM_PhysicalPackage	Optional	The physical aspects of the drive. This is required when modeling physical drives and shall not be implemented for virtual drives in virtual system environments.
31.6.5 CIM_ProtocolEndpoint	Optional	
31.6.6 CIM_Realizes	Mandatory	Associates MediaAccessDevice and PhysicalPackage.
31.6.7 CIM_SAPAvailableForElement	Conditional	Conditional requirement: Support for ProtocolEndpoints. Associates MediaAccessDevice to ProtocolEndpoint.
31.6.8 CIM_SystemDevice	Mandatory	ComputerSystem to MediaAccessDevice.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_MediaAccessDevice	Optional	MediaAccessDevice Creation. See31.3.3 Hot swap insertion or Removal of Drives.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_MediaAccessDevice	Optional	MediaAccessDevice Removal. See31.3.3 Hot swap insertion or Removal of Drives.

31.6.1 CIM_EnabledLogicalElementCapabilities

Requirement: Mandatory

Table 310 describes class CIM_EnabledLogicalElementCapabilities.

Table 310 - SMI Referenced Properties/Methods for CIM_EnabledLogicalElementCapabilities

Properties	Flags	Requirement	Description & Notes
RequestedStatesSupported		Mandatory	Possible states that can be requested when using the method RequestStateChange(). If RequestState and RequestStateChange are not implemented then RequestedStatesSupported would indicate none supported.

31.6.2 CIM_HostedAccessPoint

ComputerSystem to storage ProtocolEndpoint.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 311 describes class CIM_HostedAccessPoint.

Table 311 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to ProtocolEndpoint

31.6.3 CIM_MediaAccessDevice

Represents a tape or optical drive.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 312 describes class CIM_MediaAccessDevice.

Table 312 - SMI Referenced Properties/Methods for CIM_MediaAccessDevice

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
DeviceID		Mandatory	

Table 312 - SMI Referenced Properties/Methods for CIM_MediaAccessDevice

Properties	Flags	Requirement	Description & Notes
Name		Mandatory	
OperationalStatus		Mandatory	Shall be 2 5 6 8 10 11 (Okay or Predictive Failure or Error or Starting or Stopping or Stopped).
LocationIndicator		Mandatory	
EnabledState		Mandatory	Possible values: 2 (Enabled - drive is Spun up and online), 3 (Disabled - drive is spun down, and offline), 4 (Shutting down - drive is spinning down), 6 (Enabled but Offline - drive is spun up but offline), 10 (Starting - drive is spinning up).
RequestedState		Optional	Possible RequestedStates: 2 Enabled (Spin up drive if it was spun down and Online the drive if it was offline), 4 (Shut down - spin down drive), 6 (Offline - offline drive).
RequestStateChange()		Conditional	Conditional requirement: Support for online/offline.

31.6.4 CIM_PhysicalPackage

The physical aspects of the drive.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 313 describes class CIM_PhysicalPackage.

Table 313 - SMI Referenced Properties/Methods for CIM_PhysicalPackage

Properties	Flags	Requirement	Description & Notes
CreationClassName		Mandatory	
Manufacturer		Mandatory	The name of the organization responsible for producing the PhysicalElement.
Model		Mandatory	The name by which the PhysicalElement is generally known.
Version		Mandatory	The version of the physical element - not necessarily the same as a software/firmware version.
SerialNumber		Mandatory	
PartNumber		Mandatory	

31.6.5 CIM_ProtocolEndpoint

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 314 describes class CIM_ProtocolEndpoint.

Table 314 - SMI Referenced Properties/Methods for CIM_ProtocolEndpoint

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	

31.6.6 CIM_Realizes

Associates MediaAccessDevice and PhysicalPackage.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 315 describes class CIM_Realizes.

Table 315 - SMI Referenced Properties/Methods for CIM_Realizes

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to PhysicalPackage
Dependent		Mandatory	Reference to MediaAccessDevice

31.6.7 CIM_SAPAvailableForElement

Associates MediaAccessDevice to ProtocolEndpoint.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for ProtocolEndpoints.

Table 316 describes class CIM_SAPAvailableForElement.

Table 316 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	Reference to MediaAccessDevice
AvailableSAP		Mandatory	Reference to ProtocolEndpoint

31.6.8 CIM_SystemDevice

ComputerSystem to MediaAccessDevice.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 317 describes class CIM_SystemDevice.

Table 317 - SMI Referenced Properties/Methods for CIM_SystemDevice

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to ComputerSystem
PartComponent		Mandatory	Reference to MediaAccessDevice

EXPERIMENTAL

EXPERIMENTAL

32 Storage Enclosure Profile

32.1 Synopsis

Profile Name: Storage Enclosure (Component Profile)

Version: 1.3.0

Organization: SNIA

Central Class: Chassis

Scoping Class: ComputerSystem

Related Profiles: Table 318 describes the related profiles for Storage Enclosure.

Table 318 - Related Profiles for Storage Enclosure

Profile Name	Organization	Version	Requirement	Description
Power Supply	SNIA	1.0.1	Optional	Experimental.
Fan	SNIA	1.5.0	Optional	Experimental.
Sensors	SNIA	1.5.0	Optional	Experimental.
Disk Drive Lite	SNIA	1.6.0	Optional	
Media Access Device	SNIA	1.0.0	Optional	
Switch	SNIA	1.6.0	Optional	

The Storage Enclosure Profile describes an enclosure that houses storage components.

32.2 Description

The Storage Enclosure Profile describes an enclosure that contains storage elements (e.g., disk or tape drives) and enclosure elements (e.g., fans and power supplies). The logical aspects of the storage and enclosure elements are defined in other profiles; this profile specializes the DMTF Physical Asset Profile adding implementation details for storage enclosures. This profile supports enclosures with a single type of storage component (such as an enclosure of disks) or a mixture of different components.

The following terms apply to this profile:

- **storage elements** are CIM logical classes that relate to storage - CIM_DiskDrive, CIM_ComputerSystem (representing a disk array or switch), etc.
- **enclosure elements** are CIM logical elements that relate to enclosure service and baseboard management - fans, power supplies, sensors, etc.
- **physical elements** are CIM physical classes that map to storage or enclosure elements, and perhaps physical hardware with no logical mapping.

32.2.1 Guidelines related to Referencing Profiles

The Storage Enclosure Profile is a component profile. The autonomous referencing profile may be Array, Storage Virtualizer, or Host Hardware RAID controller. The following guidelines apply to how this profile is referenced by other profiles:

32.2.1.1 Guideline 1 - enclosure elements dedicated to a single top-level system

If the components of the enclosure are all dedicated to a single top-level System, then the profile defining that system shall be the referencing profile for the enclosure. All components (storage elements, enclosure elements, physical elements) need to be dedicated. For example, if the enclosure is used by a disk array, the CIM_ComputerSystem from the Array profile serves as the scoping instance for all the elements of the enclosure.

Note that the top-level system may be part of an autonomous profile that supports the SNIA Multiple Computer System Profile.

Note that other autonomous profiles may be dedicated as a component of another autonomous profile. For example, a Fibre Channel switch may share an enclosure with, and be dedicated as, a component of an Array.

32.2.1.2 Guideline 2 - enclosure elements shared by multiple top-level systems

If the elements of the enclosure support use by multiple top-level systems, then the referencing profile shall be the base system profile.

Examples include a JBOD array access by multiple servers or multiple switch blades sharing an enclosure.

32.2.1.3 Guideline 3 - enclosure elements need not be scoped by the system as storage elements

CIM requires instantiation of all weak associations whenever the referenced elements are instantiated. For example, every CIM_LogicalDevice instance shall be referenced by a CIM_SystemDevice association. But it is possible to have devices scope to different systems associated to each other by non-weak associations. In particular, when guideline 2 applies, enclosure elements scoped to the enclosure top-level system may be associated to storage elements scoped to a different top-level system. For example, CIM_AssociatedCooling can reference a CIM_Fan scoped to the enclosure system and a CIM_DiskDrive scoped to a server. In another example, CIM_SuppliesPower references a CIM_PowerSupply scoped to an Array within an enclosure and a CIM_ComputerSystem representing a switch.

Figure 43: "Enclosure with Two Arrays" is an example of two arrays that each have their own enclosure but share cooling. The two array enclosures are contained in an enclosure that provides a fan shared by the array elements.

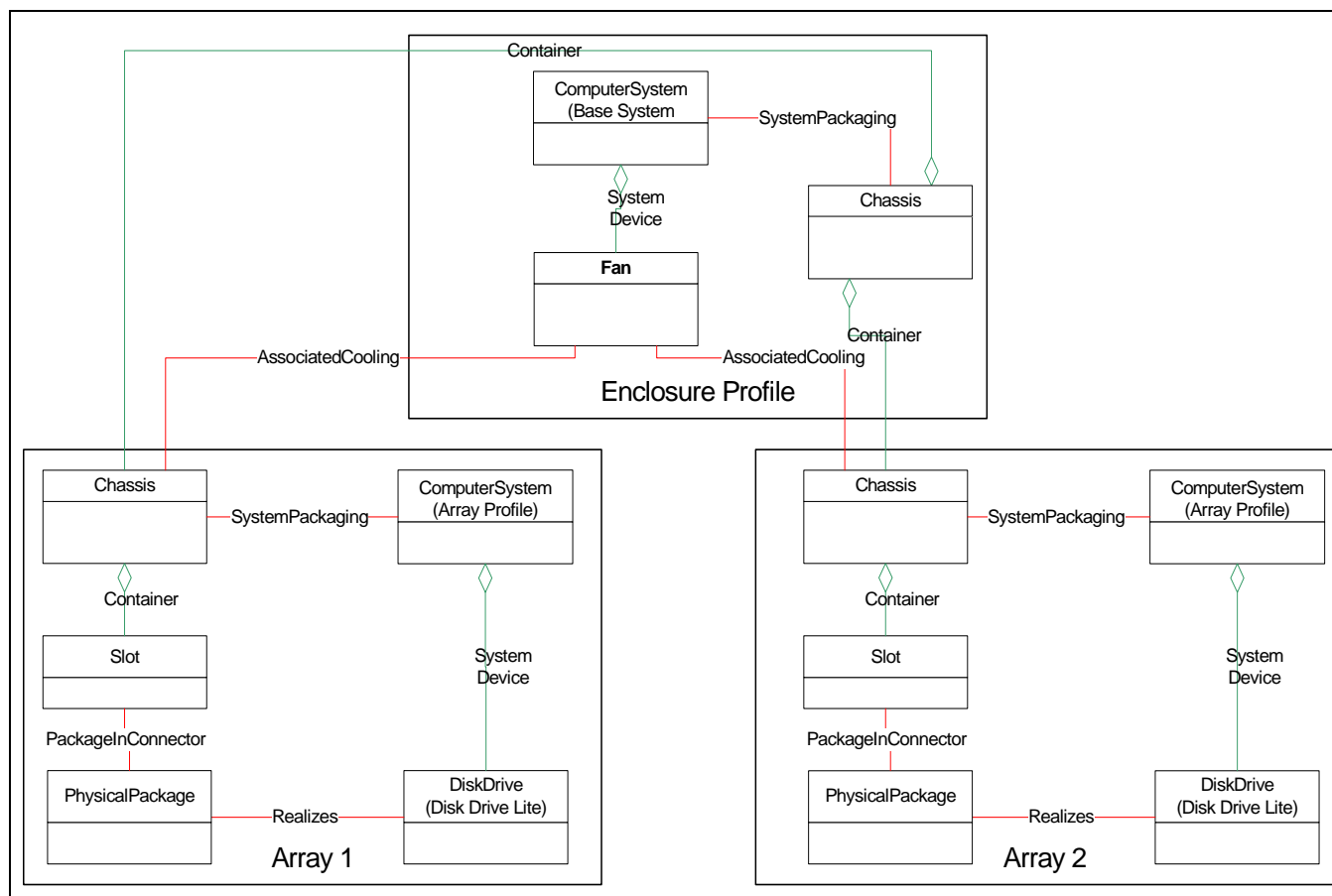


Figure 43 - Enclosure with Two Arrays

32.2.2 Examples of Storage Enclosure Configurations

32.2.2.1 Enclosure Dedicated to a Disk Array

The referencing profile is Array. Disk Drive Lite is a mandatory component profile. The physical model for disks as defined in 32.3.5.2 is mandatory.

32.2.2.2 Enclosure Dedicated to a RAID Host Controller

The referencing profile is the Host Hardware RAID Profile. Support for the Disk Drive Lite Profile is mandatory. The physical model for disks as defined in 32.3.5.2 is mandatory.

32.2.2.3 Enclosure Dedicated to non-RAID Controllers on a Single Server

The referencing profile is the Base System Profile referencing the Storage HBA Profile (or the FC HBA Profile).

32.2.2.4 Enclosure Dedicated to non-RAID Controllers on Multiple Servers

Guideline 2 applies. The referencing profile is the base system profile.

Guideline 3 may apply.

32.2.2.5 FC Switch as a Component of an Array

The Array and FC Switch share an enclosure, but the FC Switch is functionally a sub-component of the array receiving cooling and power from the enclosure. In this configuration, Array is the referencing profile to the Storage Enclosure. Guideline 3 may apply;

32.2.2.6 Enclosure containing multiple FC Switches (Director)

The enclosure is a director class switch which contains one or more switches and other devices including a FCIP Extenders and iSCSI Gateway. The referencing profile is the base system profile. Guideline 2 applies. Guideline 3 may apply.

32.3 Implementation

32.3.1 Health and Fault Management Consideration

Not defined in this standard.

32.3.2 Cascading Considerations

Not defined in this standard.

32.3.3 Enclosure Elements

32.3.3.1 Power Supplies

A storage enclosure may be modeled with one or more power supplies for device powering.

The CIM_SystemDevice association is used in the Power Supply Profile to connect the power supply to the managed system. The CIM_SuppliesPower association may be used to represent device powering to other enclosure elements of the top-level system as well as logical devices scoped to other systems.

32.3.3.2 Fans

A storage enclosure may be modeled with one or more fans for device cooling.

The CIM_SystemDevice association is used in the Fan Profile to connect the fan to the managed system. The CIM_AssociatedCooling association may be used to represent device powering to other enclosure elements of the top-level system as well as logical devices scoped to other systems.

32.3.3.3 Sensors

A storage enclosure may be modeled with one or more sensors for monitoring such factors as temperature or fan speed.

The CIM_SystemDevice association is used in the Sensors Profile to connect the sensor to the managed system. The CIM_AssociatedSensor association may be used to associate the sensor to other enclosure elements of the top-level system as well as logical devices scoped to other systems.

32.3.4 Storage Elements

32.3.4.1 Considerations for Media Access Devices in a Storage Enclosure

A storage enclosure may contain devices such as disk drives or switches. Each media access device is described by a corresponding device class as described in the corresponding profile. Each device may be associated to a physical bay or slot. The physical model for a disk drive describes a CIM_MediaAccessDevice associated to CIM_PhysicalPackage via CIM_Realizes, and CIM_Slot associated to the CIM_PhysicalPackage via CIM_PackageInConnector. If the implementation also supports hierarchical packaging, the CIM_Slot shall be associated to the CIM_PhysicalPackage realizing the referencing system or an enclosure nested in the system CIM_PhysicalPackage.

32.3.4.2 Disk Drive Considerations

If the implementation also supports the Disk Drive Lite Profile, the individual drives in the storage enclosure shall be described by an instance of CIM_DiskDrive subclassed from CIM_MediaAccessDrive. CIM_PhysicalPackage and CIM_Realizes from the Disk Drive Lite Profile shall provide the instances described in 32.3.4.1.

32.3.4.3 Media Access Devices and the Fan Profile

The Fan Profile describes fans used for device cooling and includes an AssociatedCooling association that references a CIM_ManagedSystemElement. If the implementation supports both the Fan and Disk Drive Lite Profiles, and utilizes the CIM_AssociatedCooling association, the CIM_AssociatedCooling association shall reference an instance of CIM_DiskDrive or an instance of CIM_Chassis.

32.3.4.4 Media Access Devices and the Power Supply Profile

The Power Supply Profile describes power supplies used for device powering and includes a CIM_SuppliesPower association that references a CIM_LogicalDevice. If the implementation supports both the Fan and Disk Drive Lite Profiles, and utilizes the CIM_SuppliesPower association, the CIM_SuppliesPower association shall reference an instance of CIM_DiskDrive or an instance of CIM_Chassis.

32.3.4.5 Configuration Reporting Service

The CIM_ConfigurationReportingService may be used to query for the CIM_MediaAccessDevice or CIM_LogicalPort subclasses supported within the enclosure, the supported total count and the currently installed count. In this way the total number of supported device slots, storage devices or connection ports may be retrieved. See the service method definitions in 32.4.1.

32.3.5 Physical Assets

The physical representation of the storage enclosure is mandatory. The core frame of the storage enclosure is described by CIM_Chassis.

32.3.5.1 Physical Package Hierarchy Considerations

A hierarchy of enclosures may be represented. The physical structure of a single enclosure, described by CIM_Chassis, may be associated with a variety of enclosure components and media devices. Any number of CIM_Packages may be used to group physical components. These packages may in turn be associated to one or more CIM_Chassis instances. In this case the CIM_PackageInChassis association shall be used.

32.3.5.2 Disk Drive or Media Access Device

If the implementation models slots within the enclosure, CIM_Slot shall be used to describe the slot. The instance of CIM_PhysicalPackage that describes the physical characteristics of the CIM_DiskDrive instance shall be associated to CIM_Slot by the CIM_PackageInConnector association. If the instance of

CIM_Slot is aggregated to an instance of CIM_Chassis, the CIM_ConnectorOnPackage association shall be used. Figure 44 illustrates the model.

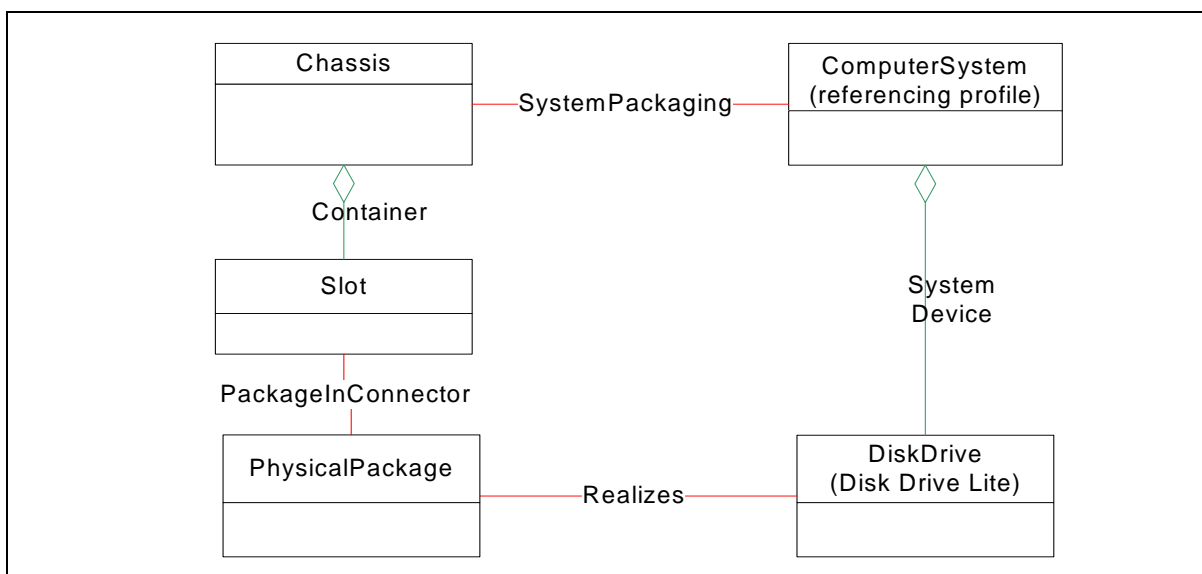


Figure 44 - Model for Disk in Enclosure

32.4 Methods

32.4.1 Extrinsic Methods of the Profile

32.4.1.1 CIM_ConfigurationReportingService GetClassTypes

GetClassTypes is used to query for the supported or currently installed device classes contained in the enclosure such as a CIM_DiskDrive or CIM_SASPort. Reporting of MediaAccessDevice derived classes directly contained within the enclosure (Recursive = False) is mandatory. Reporting of LogicalPort derived classes is optional.

The instrumentation shall support InquiryType parameter values of 2 (Supports) and 3 (Installed).

The instrumentation shall support a Recursive parameter value of false.

The instrumentation shall accept a reference to the top-level ComputerSystem in the Target parameter.

32.4.1.2 CIM_ConfigurationReportingService GetUnitTypes

GetUnitTypes is used to query for the supported or currently installed type of devices contained in the enclosure.

The instrumentation shall support InquiryType parameter values of 2 (Supports) and 3 (Installed).

The instrumentation shall support a Recursive parameter value of false.

The parameter UnitTypes may be set to "Contained", "StorageMediaLocation", "Front Side" or "Back Side". Support of the type "Contained" and "StorageMediaLocation" is mandatory. Support of "Front Side" or "Back Side" is optional. Types "Front Side" or "Back Side" are used to query for the count of the respective LogicalPorts.

32.4.1.3 CIM_ConfigurationReportingService ReportCapacity

ReportCapacity is used after GetClassTypes or GetUnitTypes is issued to find what subclasses and types are available in the enclosure, the ReportCapacity can be used to request the total supported or currently installed storage device slot count or data connection ports for the enclosure.

The instrumentation shall support InquiryType parameter values of 2 (Supports) and 3 (Installed).

The instrumentation shall support a Recursive parameter value of false.

32.4.2 Intrinsic Methods of this Profile

The profile supports read methods and association traversal. Specifically, the list of intrinsic operations supported are as follows:

- GetInstance
- Associators
- AssociatorNames
- References
- ReferenceNames
- EnumerateInstances
- EnumerateInstanceNames

32.5 Use Cases

Not in this specification

32.6 CIM Elements

Table 319 describes the CIM elements for Storage Enclosure.

Table 319 - CIM Elements for Storage Enclosure

Element Name	Requirement	Description
32.6.1 CIM_ConfigurationReportingService	Mandatory	
32.6.2 CIM_HostedService	Mandatory	Associates the CIM_ConfigurationReportingService to the System in the referencing profile.

32.6.1 CIM_ConfigurationReportingService

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 320 describes class CIM_ConfigurationReportingService.

Table 320 - SMI Referenced Properties/Methods for CIM_ConfigurationReportingService

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	
Name		Mandatory	
CreationClassName		Mandatory	
GetClassTypes()		Mandatory	
GetUnitTypes()		Mandatory	
ReportCapacity()		Mandatory	

32.6.2 CIM_HostedService

Associates the CIM_ConfigurationReportingService to the System in the referencing profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 321 describes class CIM_HostedService.

Table 321 - SMI Referenced Properties/Methods for CIM_HostedService

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	The reference to the System.
Dependent		Mandatory	The reference to the Service.

EXPERIMENTAL

STABLE

33 Software Profile

33.1 Synopsis

Profile Name: Software (Component Profile)

Version: 1.4.0

Organization: SNIA

Central Class: SoftwareIdentity

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

33.2 Description

The Software Profile models software or firmware installed on a computer system.

Information on the installed software is given using the SoftwareIdentity class. This is linked to the system using a InstalledSoftwareIdentity association.

Software information may be associated with the “top” level ComputerSystem (if all components are using the same software) or a component ComputerSystem if the software loaded can vary by processor.

Firmware is modeled as SoftwareIdentity. InstalledSoftwareIdentity is used for firmware associated with a System.

Figure 45: "Software Instance Diagram" contains the instance diagram for the Software Profile.

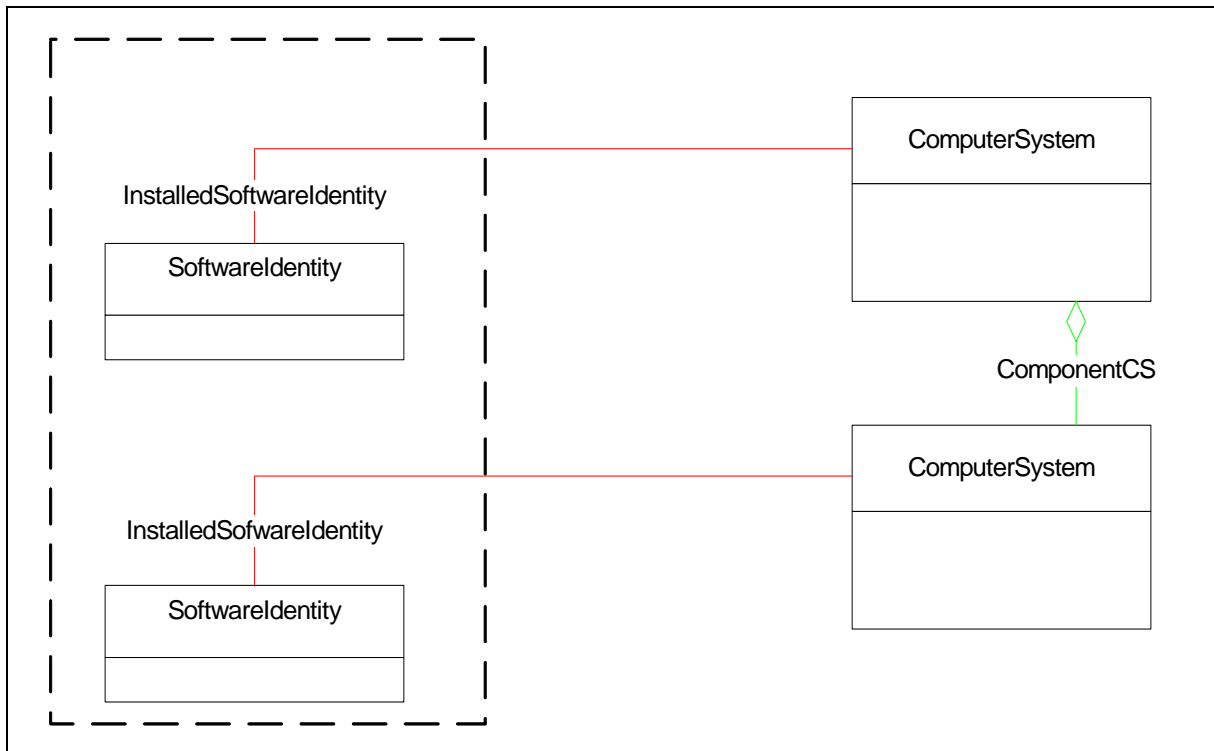


Figure 45 - Software Instance Diagram

33.3 Health and Fault Management Considerations

Not defined in this standard.

33.4 Cascading Considerations

Not defined in this standard.

33.5 Methods of the Profile

Not defined in this standard

33.6 Use Cases

Not defined in this version of the standard.

33.7 CIM Elements

Table 322 describes the CIM elements for Software.

Table 322 - CIM Elements for Software

Element Name	Requirement	Description
33.7.1 CIM_InstalledSoftwareIdentity	Mandatory	
33.7.2 CIM_SoftwareIdentity	Mandatory	

33.7.1 CIM_InstalledSoftwareIdentity

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 323 describes class CIM_InstalledSoftwareIdentity.

Table 323 - SMI Referenced Properties/Methods for CIM_InstalledSoftwareIdentity

Properties	Flags	Requirement	Description & Notes
System		Mandatory	
InstalledSoftware		Mandatory	

33.7.2 CIM_SoftwareIdentity

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 324 describes class CIM_SoftwareIdentity.

Table 324 - SMI Referenced Properties/Methods for CIM_SoftwareIdentity

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
VersionString		Mandatory	
Manufacturer		Mandatory	
BuildNumber		Optional	
MajorVersion		Optional	
RevisionNumber		Optional	
MinorVersion		Optional	

STABLE

EXPERIMENTAL

34 Software Inventory Profile

34.1 Synopsis

Profile Name: Software Inventory (Component Profile)

Version: 1.0.0

Organization: SNIA

Specializes: DMTF Software Inventory Profile 1.0.0, DSP1023 Version 1.0.1

Central Class: CIM_SoftwareIdentity

Scoping Class: a CIM_System in a referencing autonomous profile

Related Profiles: Table 325 describes the related profiles for Software Inventory.

Table 325 - Related Profiles for Software Inventory

Profile Name	Organization	Version	Requirement	Description
Indications	DMTF	1.2.2	Mandatory	See DSP1054, version 1.2.2

The Software Inventory Profile models installed and available software and firmware. The SNIA version specializes the DMTF profile in order to add indications.

34.2 Description

The Software Inventory Profile models installed and available software and firmware. The SNIA version specializes the DMTF profile in order to add indications.

34.2.1 Relationship to the SMI-S Software Profile

SMI-S defined a similar profile, the Software Profile (see 33 "Software Profile"). There are several differences between the two profiles:

- The Software Profile is limited to modeling software/firmware associated to a system and makes no provision for software/firmware associated to other elements (drives, ports,...)
- The DMTF Software Inventory Profile provides additional functionality:
 - software that is available on the system, but not installed - allowing the ability to model software/firmware that has been downloaded, but not activated.
 - collections of SoftwareIdentity instances
 - locations (such as URLs) associated with SoftwareIdentity instances

Also note that supporting this profile in SMI-S allows us to utilize the DMTF profiles which in turn use the Software Inventory Profile.

Note that although both profiles use InstalledSoftwareIdentity, the semantics are different. In the SMI-S Software Profile, InstalledSoftwareIdentity indicates that the software is both available and installed on the system. In the DMTF Software Inventory Profile, InstalledSoftwareIdentity indicates that the software is available (downloaded) on the system, and ElementSoftwareIdentity indicates that the software is

active for the referenced element. Also note that Software Inventory Profile has requirements for version properties beyond those in the SNIA Software Profile.

34.3 Implementation

See DSP1023, DMTF Software Inventory Profile.

34.3.1 Software Installation and Update

The CIM interface for Software Updates is described in the DMTF Software Update Profile (DSP1025). As a side effect of installation or updates, the inventory of software identities modeled in this profile is modified. This specialization adds indication filters:

- InstCreation of SoftwareIdentity represents a newly available software element (or new version)
- InstDeletion of SoftwareIdentity represents the deletion of an inactive SoftwareIdentity
- InstAlert with a Standard Message is used when a software (or firmware) version is updated “in-place” without installing a separate software/firmware package
- InstModification of ElementSoftwareIdentity.ElementSoftwareStatus (see 7.4.1.1 in DSP1023, DMTF Software Inventory Profile)

34.3.2 Health and Fault Management Consideration

Not defined in this standard

34.3.3 Cascading Considerations

Not defined in this standard

34.4 Methods

See DSP1023, DMTF Software Inventory Profile.

34.5 Use Cases

See DSP1023, DMTF Software Inventory Profile.

34.6 CIM Elements

Table 326 describes the CIM elements for Software Inventory.

Table 326 - CIM Elements for Software Inventory

Element Name	Requirement	Description
34.6.1 CIM_ElementSoftwareIdentity	Optional	
34.6.2 CIM_HostedAccessPoint	Optional	
34.6.3 CIM_HostedCollection	Conditional	Conditional requirement: Support for collection of SoftwareIdentity instances.
34.6.4 CIM_InstalledSoftwareIdentity	Optional	
34.6.5 CIM_MemberOfCollection	Conditional	Conditional requirement: Support for collection of SoftwareIdentity instances.
34.6.6 CIM_OrderedComponent	Optional	
34.6.7 CIM_OrderedDependency	Optional	
34.6.8 CIM_SAPAvailableForElement	Conditional	Conditional requirement: Support for SoftwareIdentityResource instances.

Table 326 - CIM Elements for Software Inventory

Element Name	Requirement	Description
34.6.9 CIM_SoftwareIdentity	Mandatory	
34.6.10 CIM_SoftwareIdentityResource	Optional	
34.6.11 CIM_SystemSpecificCollection	Optional	
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_SoftwareIdentity	Mandatory	Creation of a SoftwareIdentity. See 34.3.1 Software Installation and Update.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_SoftwareIdentity	Mandatory	Delete of a SoftwareIdentity. See 34.3.1 Software Installation and Update.
SELECT * FROM CIM_AlertIndication WHERE OwningEntity=SNIA and MessageID=\Core1\"	Mandatory	In-place update of Software (or Firmware). See 34.3.1 Software Installation and Update.
SELECT * FROM CIM_InstModification WHERE SourceInstance ISA CIM_SoftwareIdentity AND SourceInstance.CIM_SoftwareIdentity::ElementSoftwareStatus <> PreviousInstance.CIM_SoftwareIdentity::ElementSoftwareStatus	Optional	CQL -Change in ElementSoftwareStatus property of SoftwareIdentity. See 34.3.1 Software Installation and Update.

34.6.1 CIM_ElementSoftwareIdentity

CIM_ElementSoftwareIdentity is used to associate an instance of CIM_ManagedElement and an instance of CIM_SoftwareIdentity when the instance of CIM_ManagedElement is instrumented.

Requirement: Optional

Table 327 describes class CIM_ElementSoftwareIdentity.

Table 327 - SMI Referenced Properties/Methods for CIM_ElementSoftwareIdentity

Properties	Flags	Requirement	Description & Notes
ElementSoftwareStatus		Mandatory	
Antecedent		Mandatory	Reference to ManagedElement
Dependent		Mandatory	Reference to SoftwareIdentity

34.6.2 CIM_HostedAccessPoint

CIM_HostedAccessPoint is used to associate CIM_System and CIM_SoftwareIdentityResource when an instance of CIM_SoftwareIdentityResource is instrumented.

Requirement: Optional

Table 328 describes class CIM_HostedAccessPoint.

Table 328 - SMI Referenced Properties/Methods for CIM_HostedAccessPoint

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to SoftwareIdentityResource

34.6.3 CIM_HostedCollection

CIM_HostedCollection is used to associate CIM_System and CIM_SystemSpecificCollection. CIM_HostedCollection is conditional and shall be implemented when an instance of CIM_SystemSpecificCollection is instrumented.

Requirement: Support for collection of SoftwareIdentity instances.

Table 329 describes class CIM_HostedCollection.

Table 329 - SMI Referenced Properties/Methods for CIM_HostedCollection

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to SystemSpecificCollection

34.6.4 CIM_InstalledSoftwareIdentity

CIM_InstalledSoftwareIdentity is used to associate an instance of CIM_System and an instance of CIM_SoftwareIdentity. CIM_InstalledSoftwareIdentity is conditional and shall be implemented when Installed Software is modeled.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 330 describes class CIM_InstalledSoftwareIdentity.

Table 330 - SMI Referenced Properties/Methods for CIM_InstalledSoftwareIdentity

Properties	Flags	Requirement	Description & Notes
System		Mandatory	Reference to ComputerSystem
InstalledSoftware		Mandatory	Reference to SoftwareIdentity

34.6.5 CIM_MemberOfCollection

CIM_MemberOfCollection is used to associate an instance of CIM_SystemSpecificCollection and an instance of CIM_SoftwareIdentity. CIM_MemberOfCollection is conditional and shall be implemented when an instance of CIM_SystemSpecificCollection is instrumented.

Requirement: Support for collection of SoftwareIdentity instances.

Table 331 describes class CIM_MemberOfCollection.

Table 331 - SMI Referenced Properties/Methods for CIM_MemberOfCollection

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	Reference to SystemSpecificCollection
Member		Mandatory	Reference to SoftwareIdentity

34.6.6 CIM_OrderedComponent

CIM_OrderedComponent is used to associate an instance of CIM_SoftwareIdentity that represents a Software Bundle and an instance of CIM_SoftwareIdentity that represents one of the discrete software images contained in the Software Bundle.

Requirement: Optional

Table 332 describes class CIM_OrderedComponent.

Table 332 - SMI Referenced Properties/Methods for CIM_OrderedComponent

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to SoftwareIdentity that represents the bundle
PartComponent		Mandatory	Reference to SoftwareIdentity that represents a discrete image

34.6.7 CIM_OrderedDependency

CIM_OrderedDependency is used to associate an instance of CIM_SoftwareIdentity that represents an Installation Dependency and an instance of CIM_SoftwareIdentity for which the Installation Dependencies are represented.

Requirement: Optional

Table 333 describes class CIM_OrderedDependency.

Table 333 - SMI Referenced Properties/Methods for CIM_OrderedDependency

Properties	Flags	Requirement	Description & Notes
AssignedSequence		Mandatory	
Antecedent		Mandatory	Reference to SoftwareIdentity that must be installed first
Dependent		Mandatory	Reference to SoftwareIdentity that is dependent

34.6.8 CIM_SAPAvailableForElement

CIM_SAPAvailableForElement is used to associate CIM_SoftwareIdentityResource and CIM_SoftwareIdentity. CIM_SAPAvailableForElement is conditional and shall be implemented when the location information of CIM_SoftwareIdentity is represented.

Requirement: Support for SoftwareIdentityResource instances.

Table 334 describes class CIM_SAPAvailableForElement.

Table 334 - SMI Referenced Properties/Methods for CIM_SAPAvailableForElement

Properties	Flags	Requirement	Description & Notes
AvailableSAP		Mandatory	Reference to SoftwareIdentityResource
ManagedElement		Mandatory	Reference to SoftwareIdentity

34.6.9 CIM_SoftwareIdentity

CIM_SoftwareIdentity is used to represent either Installed Software or Available Software.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 335 describes class CIM_SoftwareIdentity.

Table 335 - SMI Referenced Properties/Methods for CIM_SoftwareIdentity

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
IsEntity		Mandatory	
VersionString		Optional	
MajorVersion		Conditional	Conditional requirement: No Support for SoftwareIdentity.VersionString.
MinorVersion		Conditional	Conditional requirement: No Support for SoftwareIdentity.VersionString.
RevisionNumber		Conditional	Conditional requirement: No Support for SoftwareIdentity.VersionString.

34.6.10 CIM_SoftwareIdentityResource

CIM_SoftwareIdentityResource is used to represent the location of a Software Identity, which could be used as input to the software installation service.

Requirement: Optional

Table 336 describes class CIM_SoftwareIdentityResource.

Table 336 - SMI Referenced Properties/Methods for CIM_SoftwareIdentityResource

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
InfoFormat		Mandatory	
AccessInfo		Mandatory	

34.6.11 CIM_SystemSpecificCollection

CIM_SystemSpecificCollection is used to represent a collection of Available Software.

Requirement: Optional

Table 337 describes class CIM_SystemSpecificCollection.

Table 337 - SMI Referenced Properties/Methods for CIM_SystemSpecificCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	

EXPERIMENTAL

DEPRECATED

35 Server Profile

The Server profile is deprecated. The replacement for this profile is the WBEM Server Profile.

DEPRECATED

STABLE

36 Profile Registration Profile

36.1 Synopsis

Profile Name: Profile Registration

Version: 1.0.0

Organization: SNIA

Specializes: DMTF Profile Registration 1.1.0

Central Class: RegisteredProfile

Related Profiles: Not specified in this standard.

Profile Registration Profile models the profiles registered in the object manager and the associations between registration classes and the domain classes implementing the profile.

36.2 Description

The SNIA Profile Registration Profile specializes the DMTF Profile Registration Profile adding the following classes:

- CIM_Product
- CIM_ProductSoftwareComponent

36.3 Implementation

SMI-S clients should use RegisteredProfile and ReferencedProfile in CIM operations to assure that implementations conforming to either SMI-S or DMTF profiles are discovered. ReferencedProfile associates two instances of RegisteredProfile. The DMTF Profile Registration Profile describes how the Antecedent and Dependent references should be used when one profile includes another in its supported/referenced profile list. Implementations are inconsistent in the use of these references and clients should be prepared for either approach; one technique to achieve this would be to specify NULL for Role and RemoteRole in Associator or AssociatorName operations.

The Scoping Class methodology defined in the DMTF Profile Registration Profile shall be implemented. The Central Class methodology may be implemented.

For each Profile instance, the supported component profiles should be identified via the ReferencedProfile association.

Instances of RegisteredProfile and ReferencedProfile are in the Interop namespace. The ManagedElement is in the implementation namespace.

RegisteredProfile instances are required for all SMI-S profiles, including those named as Packages.

36.3.1 ElementConformsToProfile Association

In addition, the ElementConformsToProfile association ties the RegisteredProfile for SMI-S autonomous profiles to scoping managed elements (typically ComputerSystems); these are the “top-level” objects defined in SMI-S autonomous profiles. Implementations shall support an ElementConformsToProfile association from at least one RegisteredProfile instance to the scoping instance.

A single ManagedElement may have zero or more ElementConformsToProfile associations to RegisteredProfiles. Regardless of the number of associated RegisteredProfiles, the ManagedElement

represents one set of resources. So for example, consider a ManagedElement that is a System that supports both the Array and Storage Virtualizer profiles. If one asks for the total amount of mapped capacity, the answer applies to both Array and Virtualizer and is not additive. See A.6 "Rules for Combining (Autonomous) Profiles" in *Storage Management Technical Specification, Part 2 Common Architecture, 1.7.0 Rev 5*.

36.3.2 Associations between Autonomous and Component Profile

The DMTF Profile Registration Profile requires the RegisteredProfile instances representing a profile and its supported profiles be associated via ReferencedProfile (which may be subclassed as ReferencedProfile). SMI-S has the additional requirement, that all supported profiles (whether supported directly or indirectly), are associated directly to the "top-level" autonomous profile.

For example, as shown in Figure 46, the Array Profile supports the Disk Sparing Profile which supports the Job Control. SMI-S requires both of these component profiles to be directly attached to the Array Profile instance, even though Job Control is actually a component profile of Disk Sparing. DMTF Profile Registration Profile also requires a ReferencedProfile association between the RegisteredProfiles for Disk Sparing and Job Control.

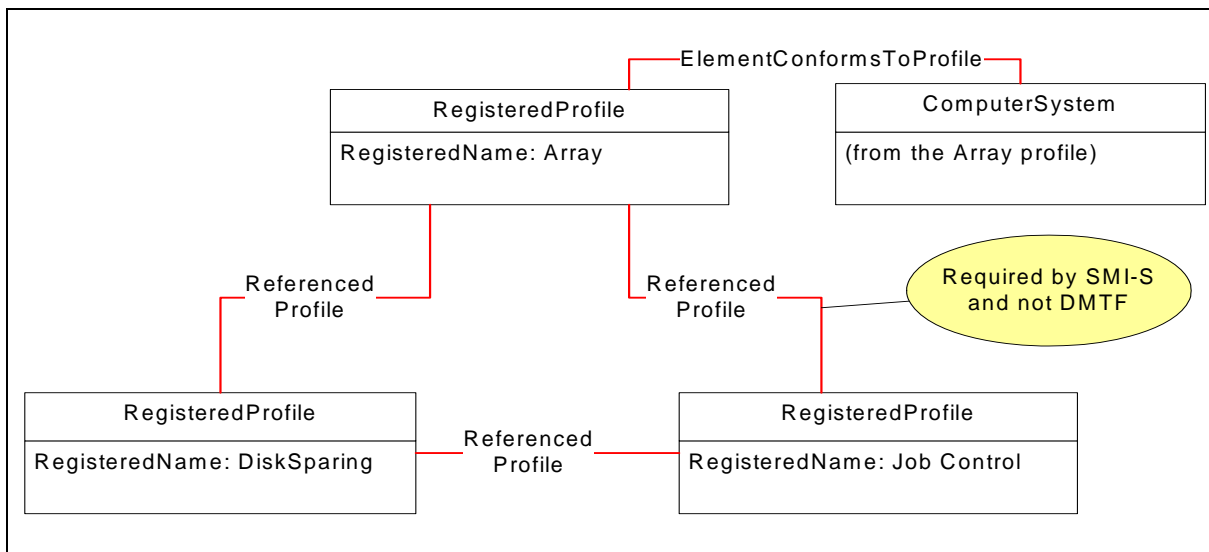


Figure 46 - Associations between RegisteredProfile instances

Each RegisteredProfile instance referenced by ElementConformsToProfile may have a set of supported profiles with RegisteredProfile instances associated using ReferencedProfile. Typically the RegisteredProfile associated via ElementConformsToProfile is for an autonomous profile and the supported profiles are component profiles. If there are multiple ElementConformsToProfile associations between a single RegisteredProfile instance and multiple domain instances, the referenced domain implementations shall support all the profiles supported by the RegisteredProfile.

36.3.3 The SMI-S Registered Profile

Each RegisteredProfile instance shall be associated to a RegisteredProfile instance holding the SMI-S version number, as shown in Figure 47. The version number (RegisteredVersion) of SMI-S profiles may or may not be the same as the version number of the SMI-S Registered Profile. The RegisteredProfile instances are associated using ElementConformsToProfile where the RegisteredProfile representing SMI storage profiles (e.g., Array, Switch) is referenced from the ManagedElement role of the association. Figure 47: "Model for SMI-S Registered Profile" depicts the RegisteredProfile representing the SMI-S

standard on the left, and RegisteredProfiles representing autonomous and component storage profiles in the middle.

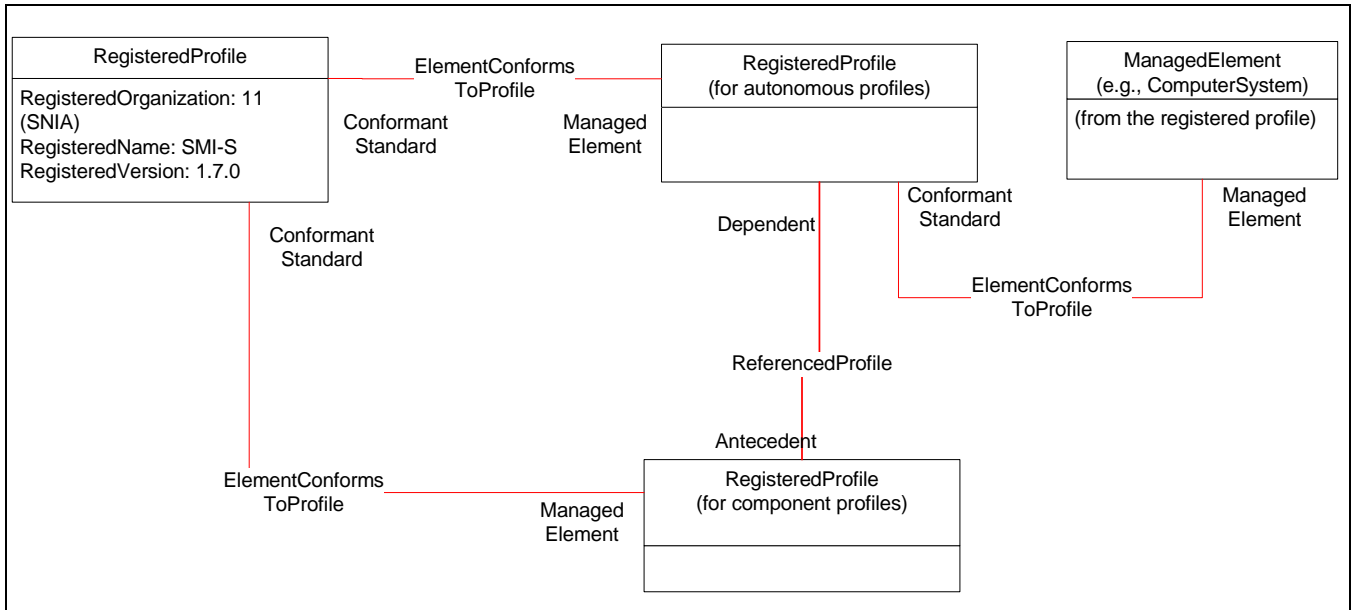


Figure 47 - Model for SMI-S Registered Profile

SMI-S class diagrams generally do not include the names of roles on associations. The requirements of roles (ConformantStandard and ManagedElement) of ElementConformsToProfile seemed critical to understand this model, so they are added to Figure 47: "Model for SMI-S Registered Profile". The role names are under the ends of the ElementConformsToProfile lines.

36.3.3.1 Provider Versions

Each RegisteredProfile instance (from the Profile Registration Profile, except the SMI-S Profile) shall be associated to one (or more) SoftwareIdentity instances containing information about the software packages required to deploy the instrumentation (including providers). These are associated using ElementSoftwareIdentity. SoftwareIdentity instance may optionally be associated to Product instances representing a software product. The model for Provider Versions is depicted in Figure 48: "Model for Provider Versions".

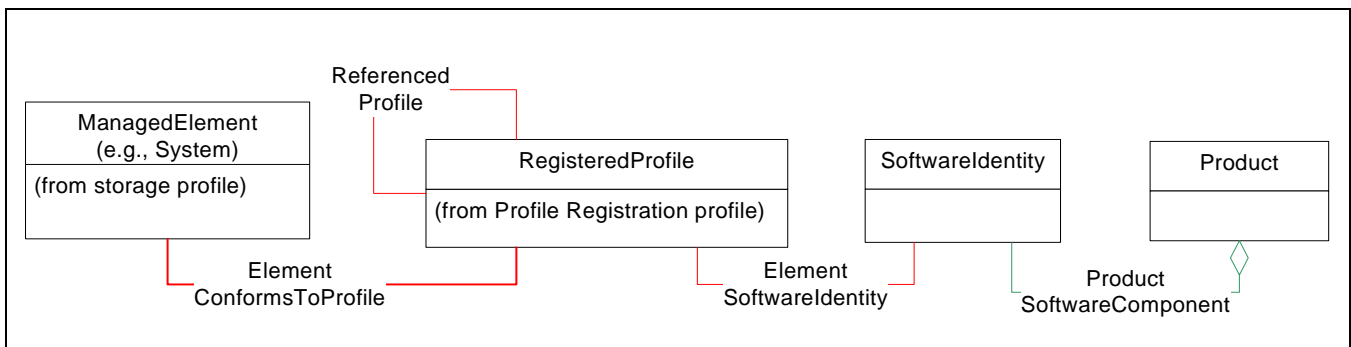


Figure 48 - Model for Provider Versions

36.3.3.2 Abstract Profile and Profile Registration

When profiles are defined for specialization, they may be defined as abstract and include this text in the Synopsis subclause:

This abstract profile specification shall not be directly implemented; implementations shall be based on a profile specification that specializes the requirements of this profile.

RegisteredProfile instances shall not be instantiated for abstract profiles. Information about abstract profiles shall not be included in the SLP template.

36.3.3.3 Indications

The Profile Registration Profile supports optional indications for the creation and deletion of RegisteredProfile instances. These indications apply to autonomous and component profiles. The indications filters are defined in terms of the RegisteredProfile class and should be triggered for RegisteredProfile as well as RegisteredProfile instances. Indications should also be triggered for creation or deletion of the SMI-S profile.

These indications will sent to subscribers when profiles are added and removed from the WBEM Server. They might be added due to updates, new functionality, or enabling licensed features. If an implementation supports dynamic creation or removal of profiles, then these indications should be supported.

36.3.4 Health and Fault Management Consideration

Not defined in this standard

36.3.5 Cascading Considerations

Not defined in this standard

36.4 Methods

Not defined in this standard

36.5 Use Cases

36.5.1 Using the WBEM Server Model to Determine SNIA Profiles Supported

All SNIA profiles require the implementation of the Server Profile as part of the WBEM Server. This allows a client to determine which SNIA profiles are supported by the a proxy, embedded or general purpose SMI-S Server. SMI-S clients can use SLP to search for services that support SNIA profiles. Indeed, a client may restrict its search to specific types of SNIA profiles. The client would get a response for each WBEM Server service that supports a SNIA profile. From the responses, the client should use the "service-id" to determine the unique WBEM Servers it is dealing with.

For each WBEM Server, the client can determine the types of entities supported by inspecting the RegisteredProfilesSupported attribute returned for the SLP entries. This identifies the types of entities (e.g., devices) supported by the WBEM Server.

The client may determine more detail on the support for the profiles by going to the service advertised for the WBEM Server and inspecting the RegisteredProfiles maintained in the server profile. This would be done by enumerating RegisteredProfiles within the interop namespace. By inspection of the actual profile instances, the client can determine the SNIA version (RegisteredVersion) of profile, associated namespaces and associated managed elements (e.g., systems).

From the RegisteredProfiles within the namespace of the ObjectManager, a client can determine other supported profiles by following the ReferencedProfile association (or its subclass ReferencedProfile). This returns a set of RegisteredProfile (or RegisteredProfile) instances that represent profiles supported by the specific autonomous profile instance. See individual profile descriptions in this specification for the specific list of "supported profiles". For a given profile instance there may be zero, one or many supported profiles.

36.5.2 Assumptions

The top-level object for each of the SMI-S profiles are:

- ComputerSystem: For Array, Storage (Media) Libraries, Virtualizers, Switches, and HBAs. This is the top-level ComputerSystem instance for the profile (not the component ComputerSystem or the member ComputerSystem);
- AdminDomain: For Fabric and HostDiscoveredResources;
- ObjectManager: For Server.

The top-level object (class instance) is associated to the RegisteredProfile instance for the profile via the ElementConformsToProfile association.

NOTE Other ManagedElement instances may be associated to the RegisteredProfile, but the meaning and behavior of such associations are not defined by SMI-S and are not mandatory.

36.6 CIM Elements

Table 338 describes the CIM elements for Profile Registration.

Table 338 - CIM Elements for Profile Registration

Element Name	Requirement	Description
36.6.1 CIM_ElementConformsToProfile (Associates Domain object (e.g. System) to RegisteredProfile)	Mandatory	Ties managed elements (e.g., Systems representing devices) to the registered profile that applies.
36.6.2 CIM_ElementConformsToProfile (Associates RegisteredProfiles for SMI-S and domain profiles)	Mandatory	Associates RegisteredProfiles for SMI-S and domain profiles.
36.6.3 CIM_ElementSoftwareIdentity (Profile and SW identity)	Mandatory	Associates a domain RegisteredProfile and SoftwareIdentity instances.
36.6.4 CIM_ElementSoftwareIdentity (Profile and SW identity)	Conditional	Conditional requirement: Support for instances of RegisteredProfile. Associates the profile and SoftwareIdentity instances.
36.6.5 CIM_Product	Optional	Represents a software product aggregating SoftwareIdentity instances with provider versions.
36.6.6 CIM_ProductSoftwareComponent	Optional	Associates Product and SoftwareIdentity.
36.6.7 CIM_ReferencedProfile	Optional	Associates referenced profiles using the DMTF Profile Registration profile.
36.6.8 CIM_RegisteredProfile (Domain Registered Profile)	Mandatory	An object representing a domain (e.g. Array or Switch) profile.
36.6.9 CIM_RegisteredProfile (The SMI-S Registered Profile)	Mandatory	A registered profile that provides the version of the SMI-S standard.
36.6.10 CIM_SoftwareIdentity	Mandatory	A representation of some bundle of providers and supporting software that shares a version number.
SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_RegisteredProfile	Optional	Creation of a registered profile instance. See 36.3.3.3 Indications.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_RegisteredProfile	Optional	Deletion of a registered profile instance. See 36.3.3.3 Indications.

36.6.1 CIM_ElementConformsToProfile (Associates Domain object (e.g. System) to RegisteredProfile)

The CIM_ElementConformsToProfile ties managed elements (e.g., Systems representing devices) to the registered profile that applies.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 339 describes class CIM_ElementConformsToProfile (Associates Domain object (e.g. System) to RegisteredProfile).

Table 339 - SMI Referenced Properties/Methods for CIM_ElementConformsToProfile (Associates Domain object (e.g. System) to RegisteredProfile)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	A element implementing a profile (e.g., top-level system).
ConformantStandard		Mandatory	RegisteredProfile instance describing the domain profile.

36.6.2 CIM_ElementConformsToProfile (Associates RegisteredProfiles for SMI-S and domain profiles)

Associates RegisteredProfiles for SMI-S and domain profiles.

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 340 describes class CIM_ElementConformsToProfile (Associates RegisteredProfiles for SMI-S and domain profiles).

Table 340 - SMI Referenced Properties/Methods for CIM_ElementConformsToProfile (Associates RegisteredProfiles for SMI-S and domain profiles)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	The RegisteredProfile representing the domain profile.
ConformantStandard		Mandatory	The SMI-S RegisteredProfile.

36.6.3 CIM_ElementSoftwareIdentity (Profile and SW identity)

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 341 describes class CIM_ElementSoftwareIdentity (Profile and SW identity).

Table 341 - SMI Referenced Properties/Methods for CIM_ElementSoftwareIdentity (Profile and SW identity)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to SoftwareIdentity.
Dependent		Mandatory	Reference to domain RegisteredProfile.

36.6.4 CIM_ElementSoftwareIdentity (Profile and SW identity)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Support for instances of RegisteredProfile.

Table 342 describes class CIM_ElementSoftwareIdentity (Profile and SW identity).

Table 342 - SMI Referenced Properties/Methods for CIM_ElementSoftwareIdentity (Profile and SW identity)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to Software Identity.
Dependent		Mandatory	Reference to RegisteredProfile.

36.6.5 CIM_Product

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 343 describes class CIM_Product.

Table 343 - SMI Referenced Properties/Methods for CIM_Product

Properties	Flags	Requirement	Description & Notes
Name		Mandatory	Commonly used product name.
IdentifyingNumber		Mandatory	Software serial number.
Vendor		Mandatory	Product supplier.
Version		Mandatory	Product version information.

36.6.6 CIM_ProductSoftwareComponent

Associates Product and SoftwareIdentity.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 344 describes class CIM_ProductSoftwareComponent.

Table 344 - SMI Referenced Properties/Methods for CIM_ProductSoftwareComponent

Properties	Flags	Requirement	Description & Notes
GroupComponent		Mandatory	Reference to Product.
PartComponent		Mandatory	Reference to SoftwareIdentity.

36.6.7 CIM_ReferencedProfile

Associates referenced profiles using the DMTF Profile Registration profile.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 345 describes class CIM_ReferencedProfile.

Table 345 - SMI Referenced Properties/Methods for CIM_ReferencedProfile

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to autonomous RegisteredProfile
Dependent		Mandatory	Reference to component RegisteredProfile

36.6.8 CIM_RegisteredProfile (Domain Registered Profile)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 346 describes class CIM_RegisteredProfile (Domain Registered Profile).

Table 346 - SMI Referenced Properties/Methods for CIM_RegisteredProfile (Domain Registered Profile)

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	This is a unique value for the profile instance.
RegisteredOrganization		Mandatory	This is the official value of the organization that created the Profile. For SMI-S profiles, this would be SNIA (11). For DMTF profiles, this would be DMTF (2).
OtherRegisteredOrganization		Conditional	Conditional requirement: CIM_RegisteredProfile requires the OtherRegisteredOrganization property be populated if the RegisteredOrganization property has a value of 1 (\Other\).'Mandatory if RegisteredOrganization is 1 ('Other').
RegisteredName		Mandatory	This is the name assigned by the organization that created the profile.
RegisteredVersion		Mandatory	This is the version number assigned by the organization that defined the Profile.

Table 346 - SMI Referenced Properties/Methods for CIM_RegisteredProfile (Domain Registered Profile)

Properties	Flags	Requirement	Description & Notes
AdvertiseTypes	N	Mandatory	Defines the advertisement of this profile. If the property is null then no advertisement is defined. A value of 1 is used to indicate 'other' and a 3 is used to indicate 'SLP'.
AdvertiseTypeDescriptions		Conditional	Conditional requirement: CIM_RegisteredProfile requires the AdvertiseTypeDescriptions property be populated if the AdvertiseTypes property has a value of 1 (\Other\). This shall not be NULL if 1 ('Other') is identified in AdvertiseType.

36.6.9 CIM_RegisteredProfile (The SMI-S Registered Profile)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 347 describes class CIM_RegisteredProfile (The SMI-S Registered Profile).

Table 347 - SMI Referenced Properties/Methods for CIM_RegisteredProfile (The SMI-S Registered Profile)

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	A unique value for the profile instance.
RegisteredOrganization		Mandatory	Shall be 11 (SNIA).
RegisteredName		Mandatory	Shall be 'SMI-S'.
RegisteredVersion		Mandatory	The version number of the SMI specification the associated profiles conform to.
AdvertiseTypes		Mandatory	Should be 2 (Not Advertised) or 3 (SLP). 2 is recommended to avoid increasing size of SLP template.
AdvertiseTypeDescriptions		Conditional	Conditional requirement: CIM_RegisteredProfile requires the AdvertiseTypeDescriptions property be populated if the AdvertiseTypes property has a value of 1 (\Other\). This shall not be NULL if 'Other' is identified in AdvertiseType.

36.6.10 CIM_SoftwareIdentity

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 348 describes class CIM_SoftwareIdentity.

Table 348 - SMI Referenced Properties/Methods for CIM_SoftwareIdentity

Properties	Flags	Requirement	Description & Notes
Name		Mandatory	A user-friendly name for the instrumentation software.
InstanceID		Mandatory	
VersionString		Mandatory	
Manufacturer		Mandatory	The name of the company associated with the instrumentation software.
Classifications		Mandatory	
ClassificationDescriptions		Conditional	Conditional requirement: CIM_SoftwareIdentity requires the ClassificationDescriptions property be populated if the Classifications property has a value of 1 ('Other'). Mandatory if Classifications is set to 1 ('Other').

STABLE

DEPRECATED

37 Indication Profile

The functionality of the Indication Profile has been replaced by the Indications Profile (42).

The Indication Profile is defined in Clause 42 of SMI-S 1.6.1.

DEPRECATED

Indication Profile

DEPRECATED

38 Object Manager Adapter Profile

The functionality of the Object Manager Adapter Profile has been replaced by the WBEM Server Profile.

The Object Manager Adapter Profile is defined in Clause 44 of SMI-S 1.6.

DEPRECATED

DEPRECATED

39 Proxy Server System Management Profile

See the last version of this profile in *SMI-S Version 1.6.1 Revision 5*.

DEPRECATED

STABLE

40 Device Credentials Profile

40.1 Synopsis

Profile Name: Device Credentials (Component Profile)

Version: 1.3.0

Organization: SNIA

Central Class: SharedSecretService

Scoping Class: a ComputerSystem in a separate autonomous profile

Related Profiles: Not defined in this standard.

40.2 Description

Many devices require a shared secret to be provided to access them. This shared secret is different that the credentials used by the SMI-S Client for authentication with the CIM Server. This profile is used to change this device shared secrets.

The SMI-S Client shall not be provided with the password, only the principle. The SMI-S Client can use the principle to change the shared secret appropriately.

The device credentials can be exposed throughout the CIM model such that a CIM Client may manipulate them. The credentials are modeled as shared secrets.

40.2.1 Instance Diagram

Figure 49: "DeviceCredentials Profile Model" provides a sample instance diagram.

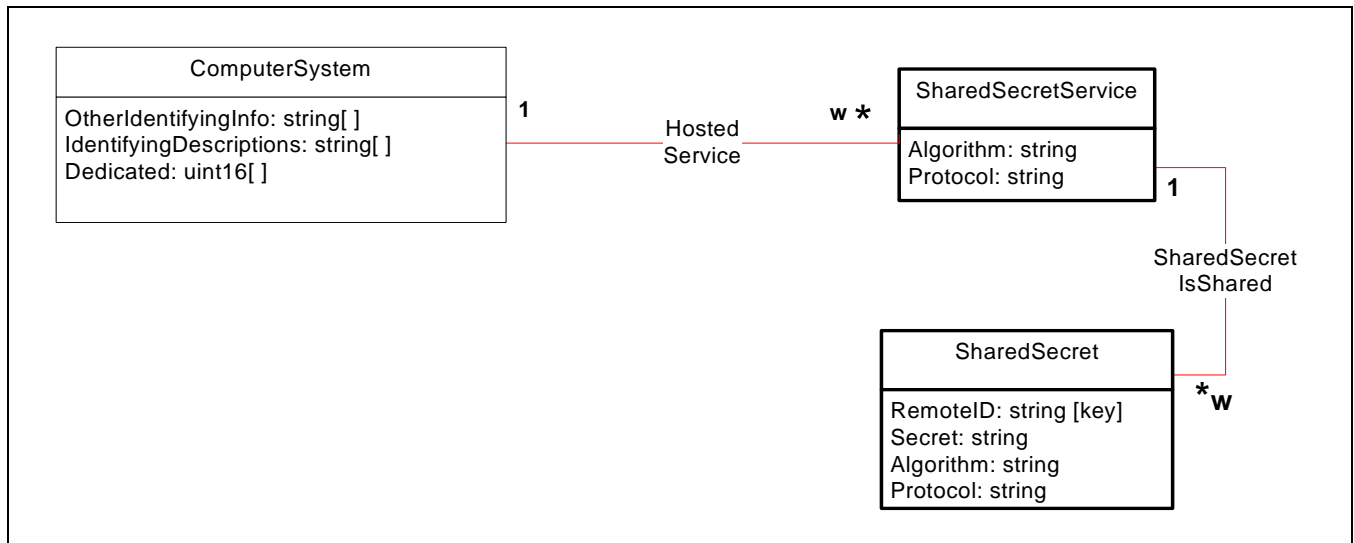


Figure 49 - DeviceCredentials Profile Model

40.3 Health and Fault Management Considerations

Not defined in this standard.

40.4 Cascading Considerations

Not defined in this standard.

40.5 Extrinsic Methods of this Profile

Not defined in this standard.

40.6 Use Cases

Not defined in this version of the standard.

40.7 CIM Elements

Table 349 describes the CIM elements for Device Credentials.

Table 349 - CIM Elements for Device Credentials

Element Name	Requirement	Description
40.7.1 CIM_HostedService	Mandatory	
40.7.2 CIM_SharedSecret	Mandatory	
40.7.3 CIM_SharedSecretIsShared	Mandatory	
40.7.4 CIM_SharedSecretService	Mandatory	

40.7.1 CIM_HostedService

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 350 describes class CIM_HostedService.

Table 350 - SMI Referenced Properties/Methods for CIM_HostedService

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to ComputerSystem
Dependent		Mandatory	Reference to SharedSecretService

40.7.2 CIM_SharedSecret

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 351 describes class CIM_SharedSecret.

Table 351 - SMI Referenced Properties/Methods for CIM_SharedSecret

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	

Table 351 - SMI Referenced Properties/Methods for CIM_SharedSecret

Properties	Flags	Requirement	Description & Notes
ServiceCreationClassName		Mandatory	
ServiceName		Mandatory	
RemotelID		Mandatory	
Secret		Mandatory	

40.7.3 CIM_SharedSecretIsShared

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 352 describes class CIM_SharedSecretIsShared.

Table 352 - SMI Referenced Properties/Methods for CIM_SharedSecretIsShared

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	Reference to SharedSecretService
Dependent		Mandatory	Reference to SharedSecret

40.7.4 CIM_SharedSecretService

Created By: Static
 Modified By: Static
 Deleted By: Static
 Requirement: Mandatory

Table 353 describes class CIM_SharedSecretService.

Table 353 - SMI Referenced Properties/Methods for CIM_SharedSecretService

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
ElementName		Mandatory	

STABLE

EXPERIMENTAL

41 Operational Power Profile

41.1 Synopsis

Profile Name: Operational Power (Component Profile)

Version: 1.7.0

Organization: SNIA

Central Class: OperationalPowerStatisticsService

Scoping Class: ComputerSystem

Related Profiles: Table 354 describes the related profiles for Operational Power.

Table 354 - Related Profiles for Operational Power

Profile Name	Organization	Version	Requirement	Description
Disk Drive Lite	SNIA	1.7.0	Optional	
Fan	SNIA	1.5.0	Optional	
Power Supply	SNIA	1.0.1	Optional	
Multiple Computer System	SNIA	1.2.0	Optional	
CPU	DMTF	1.0.0	Optional	
Device Tray	DMTF	1.0.0	Optional	

41.2 Description

41.2.1 Overview

The Operational Power Profile defines classes and methods for viewing system power usage information in real time. It is a component profile supported by autonomous profiles such as the Array and Self-Contained NAS Profile.

Emerging data center best practices require close monitoring of the energy used by various system components. These include the CPUs, chipsets, fans, power supplies, disks and PDUs (Power Distribution Units) used in storage systems.

Systems have various capabilities with respect to the granularity of information they are able to provide. For this reason, nearly all the classes and properties in this profile, except for whole-system info, are listed as optional. However, implementers must understand that the more granular the information that a data center manager can obtain, the more specifically they can tune their power and air conditioning systems. So it is advantageous to implement every property on which the underlying system supports reporting.

This profile is specifically patterned after the Block Server Performance and the Filesystem Performance Profiles in order to ease implementation by both client and server-side developers already experienced with those profiles.

41.3 Implementation

41.3.1 Model Overview

Figure 50: "Operational Power Profile Summary" provides an overview of the model. The ComputerSystem is that of the autonomous profile (e.g., a NAS Head or a Self-Contained NAS) which utilizes the Operational Power Profile. This set of classes indicates the system supports power statistics for the entire system (OperationalPowerStatistics with ElementType = 2) and disks (OperationalPowerStatistics with ElementType = 10). There may be multiple instances of OperationalPowerStatistics with ElementType = 2 providing power statistics for multiple disks. There may be power statistics for other element types (see 41.3.2 Element Types).

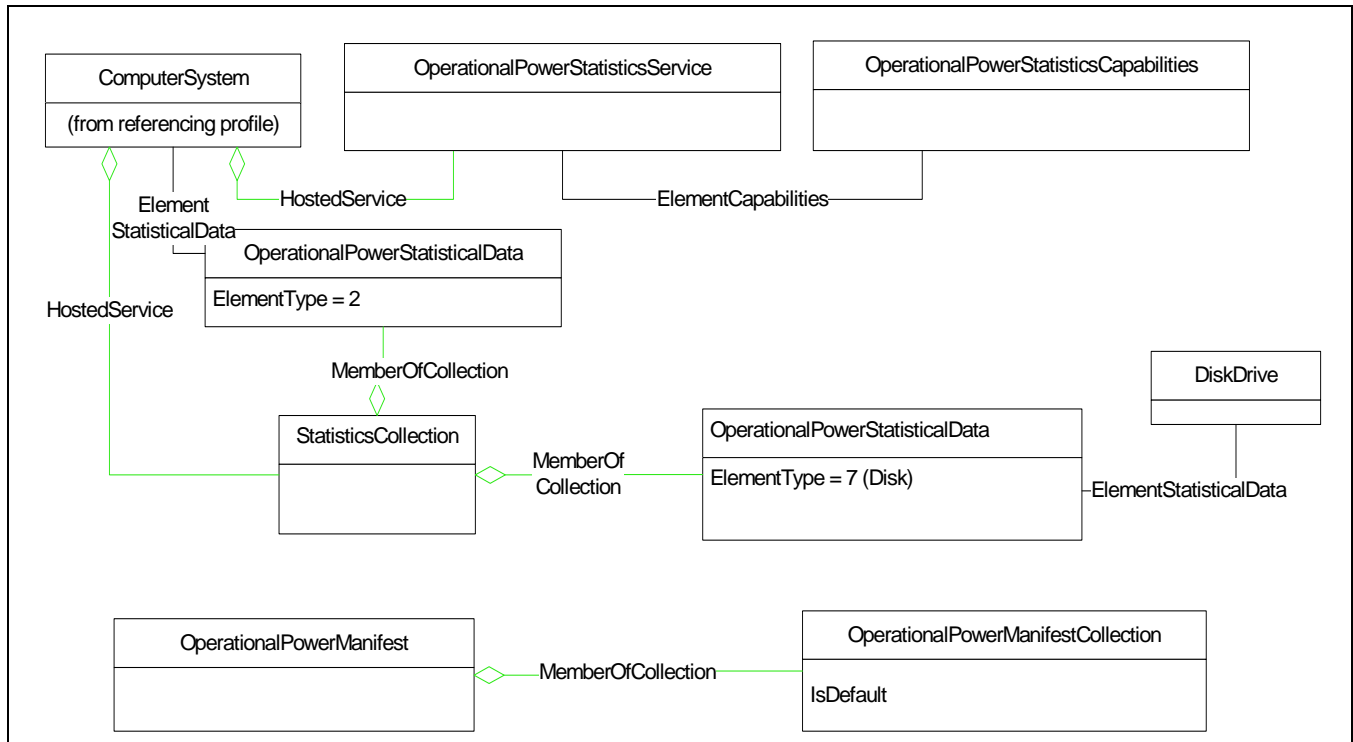


Figure 50 - Operational Power Profile Summary

The StatisticsCollection is the anchor point from which all statistics being kept by the profile can be found. Statistics are defined as an OperationalPowerStatisticalData class, instances of which hold the statistics for particular metered elements (i.e., whole systems, PDUs, power supplies, disks, disk trays, RAID groups, and fans). The particular type of metered element is recorded in the instance of OperationalPowerStatisticalData within the ElementType property.

All of the statistics instances are related to the elements that they meter via the ElementStatisticalData association (e.g., OperationalPowerStatisticalData for a DiskDrive can be found from the DiskDrive by traversing the ElementStatisticalData association).

All of the statistics instances kept within the profile are associated to the one StatisticsCollection instance. Access to all of the statistics for the profile is through the StatisticsCollection. The StatisticsCollection has a HostedCollection association to the "top level" computer system of the profile.

41.3.2 Element Types

Statistics may be kept for a number of elements within the profile, including elements within other component profiles. For example, power statistics may be associated with an instance of DiskDrive

implemented as part of the SMI-S Disk Drive Lite Profile. If the implementation does not support the Disk Drive Live Profile, then it may chose to support DiskDrive instances purely for this profile.

Figure 51, "Model for Element Types" provides an overview of the models for various element types.

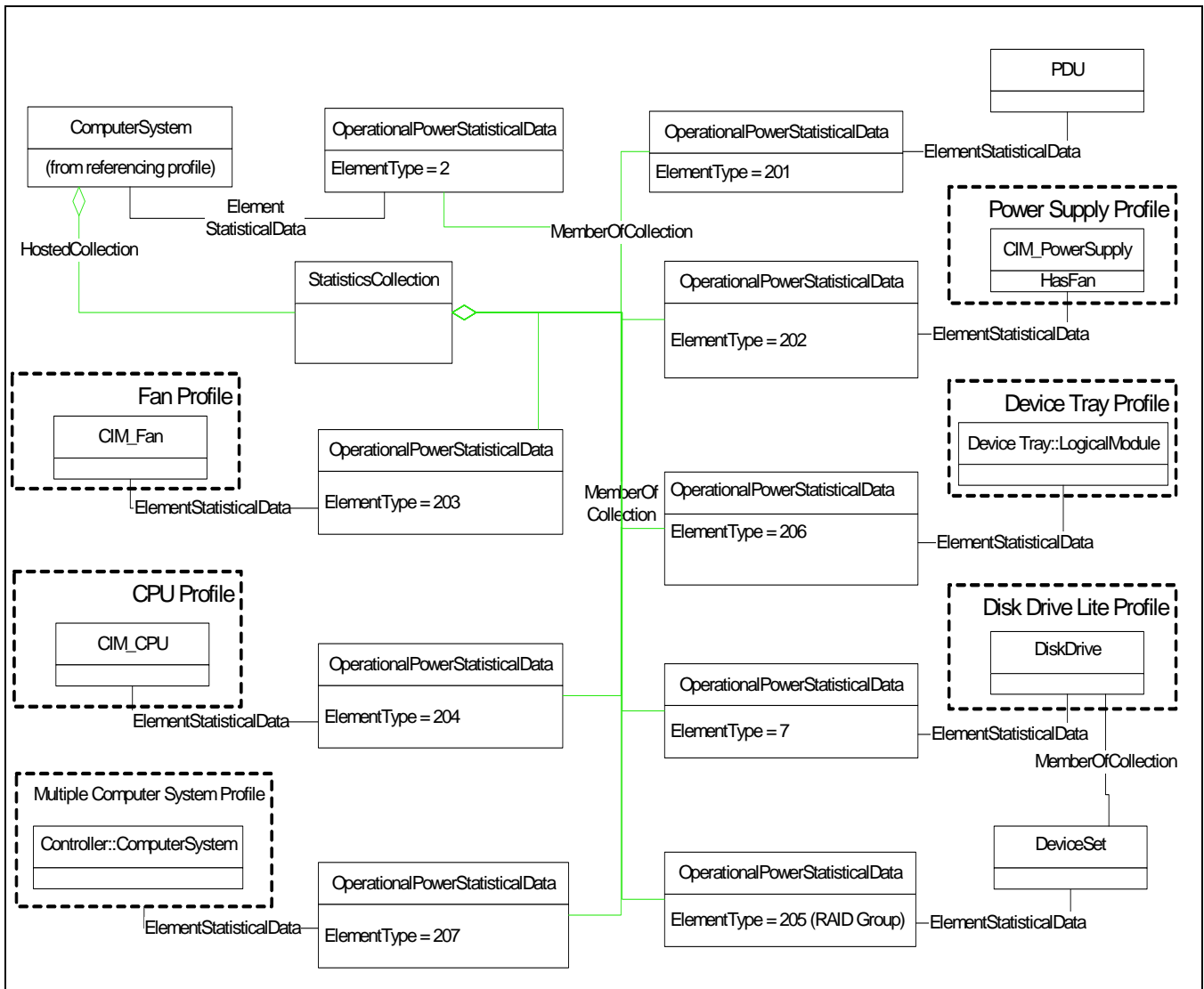


Figure 51 - Model for Element Types

OperationalPowerStatisticsCapabilities.ElementsSupported holds a list of the types of elements supported by this profile. See 41.3.7.1 **ElementsSupported**.

41.3.2.1 "Top Level" System

The top-level system is the **ComputerSystem** instance defined in an autonomous profile (e.g., Array, NAS Head) that supports the Operational Power Profile. This top-level system represents the entire storage system; power statistics for the top-level system represent the aggregation of values from all elements within the system. In other words, the milliwatt value reported for the top-level statistics should equal the sum of the Milliwatts reported for other elements. Power statistics for the top-level system are mandatory. Figure 52, "Classes related to Top-level System Power Statistics" which is a subset of the model from

Figure 51, “Model for Element Types” shows the mandatory classes for power statistics for the top-level system.

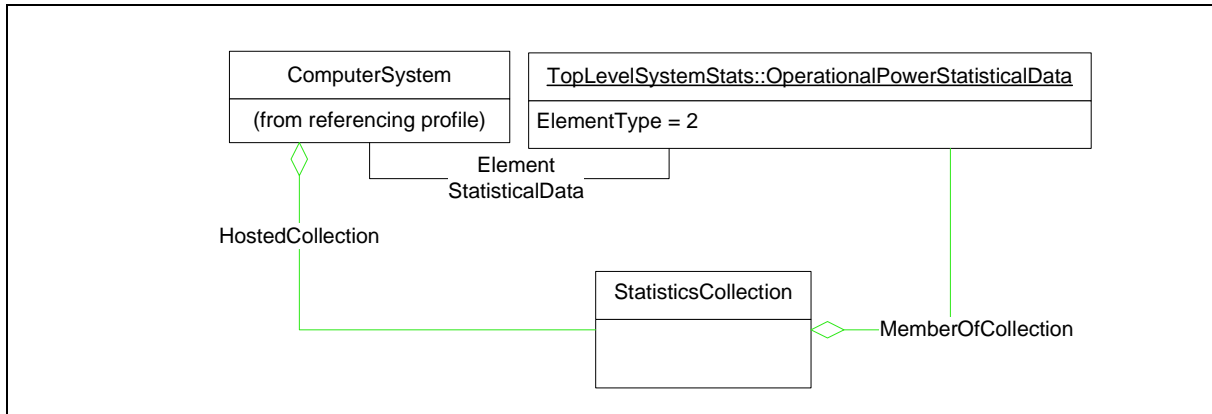


Figure 52 - Classes related to Top-level System Power Statistics

41.3.2.2 Power Source Statistics

Within this profile, a power source that provides statistics and provides some source of power conversion (for example, from AC to DC) is known as a Power Supply and is modeled using CIM_PowerSupply. A device that provides power statistics without information about conversions is known as a Power Distribution Unit (PDU) and is modeled using CIM_PowerSource. Any given subset of power may be represented by either a PDU (CIM_PowerSource) or Power Supply (CIM_PowerSupply), but shall not be represented by both.

The power statistics for Power Supplies or PDUs are kept within the OperationalPowerStatisticalData instances, with one for each Power Supply or PDU within the system. The Milliwatts attribute of OperationalPowerStatisticalData associated to PowerSource or PowerSupply represents the power dedicated to that power source, not the power supplied by it. In other words, it represents the overhead of the power source. There may be multiple instances per disk tray or controller. The implementation may use the DMTF Power Supply Profile (DSP1015) or model key classes as follows:

Each PowerSupply or PowerSource instance shall be associated to the ComputerSystem using SystemDevice.

If no instance of CIM_SuppliesPower references the instance of CIM_PowerSupply (or PowerSource), the power supply represented by CIM_PowerSupply or PowerSource supplies power to the whole managed system. In this case, the CIM_ComputerSystem instance and the CIM_PowerSupply or PowerSource instance shall only be associated through an instance of CIM_SystemDevice.

If at least one instance of CIM_SuppliesPower references the instance of CIM_PowerSupply (or PowerSource), all of the power-receiving elements shall be associated with the CIM_PowerSupply (or PowerSource) instance through an instance of CIM_SuppliesPower.

The ElementType associated with a PDU is 201; the ElementType associated with a power supply is 202.

41.3.2.3 Disk tray

Each disk tray is represented by a separate OperationalPowerStatisticalData instance. The tray itself is modeled by an instance of CIM_LogicalModule. The implementation may use the DMTF Device Tray Profile (DSP1019) or may model key classes as follows:

Each LogicalModule instance shall be associated to the ComputerSystem using SystemDevice. There shall be ConcreteComponent associations between LogicalModule and contained disks. The Milliwatts

value in the `OperationalPowerStatisticalData` instance associated with `LogicalModule` shall be the sum of values of disks residing in the tray.

The `ElementType` associated with a disk tray is 206.

41.3.2.4 Fan

The power statistics for each fan may be presented in a separate `OperationalPowerStatisticalData` instance. There may be multiple instances per disk tray or controller head. The implementation may use the DMTF Fan Profile (DSP1013) or may model key classes as follows:

Each Fan instance shall be associated to the `ComputerSystem` using `SystemDevice`.

The `ElementType` associated with a fan is 203.

41.3.2.5 CPU

The power statistics for each CPU in a system may be modeled by a separate `OperationalPowerStatisticalData` instance. At the provider's discretion, each core on the system may have its own instance. While the naming of these instances is not specified, there may be a need to indicate whether cores or entire CPU units are being reported. The implementation may use the DMTF CPU Profile (DSP1022) or may model key classes as follows:

Each CPU (`CIM_Processor`) instance shall be associated to the `ComputerSystem` using `SystemDevice`.

The `ElementType` associated with a CPU is 204.

41.3.2.6 Disk Drive

The statistics for each disk drive in a system is represented by a separate `OperationalPowerStatisticalData` instance. The implementation may use the SNIA Disk Drive Lite Profile or may model key classes as follows:

Each Disk Drive (`CIM_DiskDrive`) instance shall be associated to the `ComputerSystem` using `SystemDevice`.

The `ElementType` associated with a disk drive is 10. This value is consistent with the value in the Block Server Performance Profile.

41.3.2.7 RAID Groups

The statistics for each RAID Group in a system is represented by a separate `OperationalPowerStatisticalData` instance. This `OperationalPowerStatisticalData` instance shall have an association to the `CIM_DeviceSet` instance representing the RAID Group. Each `DiskDrive` instance which is a member of the RAID group shall be associated to the `CIM_DeviceSet` via `MemberOfCollection`. The Milliwatts value in `OperationalPowerStatisticalData` representing the RAID group shall be the sum of the values of the associated disks.

The `ElementType` associated with a RAID Group is 205.

41.3.2.8 Controller

If the implementation supports the SNIA Multiple Computer System Profile, it may model controller statistics with `OperationalPowerStatisticalData` instances associated to `ComputerSystem` instances representing controllers.

The `ElementType` associated with a RAID Group is 207.

41.3.3 Power Metric Attributes

At this time, the following attributes are defined for all Elements described in 41.3.2 "Element Types":

- **Milliwatt:** the current value in Milliwatts of the power consumed by the element referenced. Note that in this standard, a milliwatt is defined to be one thousandth of a watt.
- **Precision:** the number of decimal places of accuracy of the Milliwatt attribute.
- **StatisticTime:** the date and time the Milliwatt value was acquired from the device. This is stored in CIM datetime format.

All three of these values are mandatory for the OperationalPowerStatisticalData instance reference the top-level ComputerSystem and optional for other elements.

41.3.4 Bulk Retrieval

Figure 50, “Operational Power Profile Summary” illustrates the OperationalPowerStatisticsService for bulk retrieval of all the statistics data and the creation of manifest collections. Associated with the OperationalPowerStatisticsService is an OperationalPowerStatisticsCapabilities instance that identifies the specific capabilities implemented by the operational power statistics support. Specifically, it includes an “ElementsSupported” property that identifies the elements for which statistics are kept; the OperationalPowerStatisticsCapabilities instance also identifies the various retrieval mechanisms (e.g., Extrinsic or Association Traversal) that are implemented (i.e., supported) by the provider’s statistics support.

41.3.5 Default Manifest Collection

Associated with the instances of the StatisticsCollection shall be a provider-supplied (Default) OperationalPowerManifestCollection that represents the statistics properties that are kept by the profile. The default manifest collection is indicated by the IsDefault property (=True) of the OperationalPowerManifestCollection. For each metered object (element) of the profile implementation, the default manifest collection will have exactly one manifest that will identify which properties are included for that metered object. If an object is not metered, then there shall not be a manifest for that element type. If an element type (e.g., CPU) is metered, then there shall be a manifest for that element type.

EXPERIMENTAL

Each default manifest in the default manifest collection identifies the properties included by default by the implementation. The CSVSequence property of the manifest shall identify the default sequence in which the implementation will return properties within each record for the ElementType on a GetStatisticsCollection request. For each property included in the manifest by the value “true”, there should be an entry in the CSVSequence array identifying the OperationalPowerStatisticalData property by name. The first three values of CSVSequence shall be "InstanceID", "ElementType" and "StatisticsTime" to allow correlation of the Manifest with the CSV record based on the ElementType.

EXPERIMENTAL

41.3.6 Client Defined Manifest Collection

Manifest collections are either provider-supplied (OperationalPowerManifestCollection.IsDefault=True) or client-defined collections (OperationalPowerManifestCollection.IsDefault=False). Client-defined collections are used to indicate the specific statistics properties that the client would like to retrieve using the GetStatisticsCollection method. For a discussion of provider-supplied manifest collections, see 41.3.5.

Client-defined manifest collections are a mechanism for restricting the amount of data returned on a GetStatisticsCollection request. A client-defined manifest collection is identified by the IsDefault property of the collection set to False. For each element type of the operational power statistics class (e.g., PDU,

Fan, etc.), a manifest can be defined that identifies which specific properties of the particular statistics class element type are to be returned on a `GetStatisticsCollection` request. Each of the element types of the operational power statistics class may have no or one manifest in any given manifest collection. This is illustrated in Figure 50: "Operational Power Profile Summary".

In Figure 50: "Operational Power Profile Summary", manifest classes are defined for PDUs, power supplies, fans, CPUs, disks and disk trays. Each property of the manifest is a Boolean that indicates whether the property is to be returned (true) or omitted (false).

Multiple client-defined manifest collections can be defined in the profile. Consequently, different clients or different client applications can define different manifests for different application needs. A manifest collection can completely omit a whole set of statistics pertaining to one or more element types; for example, one manifest might collect information on disk power only. Since manifest collections are "client objects", they are named (ElementName) by the client for the client's convenience. The CIM server will generate an instance ID to uniquely identify the manifest collection in the CIM Server.

Client-defined manifest collections are created using the `CreateManifestCollection` method. Manifests are added or modified using the `AddOrModifyManifest` method. A manifest may be removed from the manifest collection by using the `RemoveManifests` method.

NOTE Use of manifest collections is optional with the `GetStatisticsCollection` method. If NULL for the manifest collection is passed on input, then all statistics instances are assumed (i.e., all available statistics will be returned).

41.3.7 Capabilities Support for Operational Power Profile

To determine if an autonomous profile (e.g. Array, NAS, Head, Self Contained NAS, ...) supports the Operational Power Profile, start with the `RegisteredProfile` instance that represents the autonomous profile, traverse the `ReferencedProfile` association to see if any of the `RegisteredProfile` instances returned represent the Operational Power Profile. In order to support statistics for a particular class of metered element, the corresponding object must be modeled. This profile requires support for all classes used by the profile, regardless of whether statistics on them are available or not. To find out whether statistics are reported, examine the `ElementsSupported` property in the `OperationalPowerStatisticsCapabilities` instance associated to the `OperationalPowerStatisticsService`. This capabilities class instance is not created nor modified by clients; rather, it is populated by the provider and has three properties of interest (as discussed within the following sections).

For the methods-supported properties described below (namely, `SynchronousMethodsSupported` and `AsynchronousMethodsSupported`), any or all of the respective values can be missing (e.g., the arrays can be NULL). If all of the methods supported are NULL, then manifest collections are not supported and `GetStatisticsCollection` is not supported for the retrieval of statistics. This leaves enumerations or association traversals as the only methods for retrieving the statistics.

41.3.7.1 ElementsSupported

This property within the `OperationalPowerStatisticsCapabilities` class defines a list of element types for which statistical data is available. The valid values are "PDU", "Disk tray", "Disk", "CPU", "Fan", "Power Supply", "RAID Group", and "Controller".

To be a valid implementation of the Operational Power Profile, at least one of the values listed for `ElementsSupported` shall be supported. `ElementsSupported` is an array, such that all of the values can be identified.

Some of these elements are available in hybrid configurations. Two common examples are multi-core CPUs and power supplies with integrated fans. Clients must examine the corresponding class instances to find out whether or not they are so configured.

41.3.7.2 SynchronousMethodsSupported

This property within the OperationalPowerStatisticsCapabilities class defines the synchronous mechanisms that are supported for retrieving statistics and for defining and modifying filters for statistics retrieval. For this release of SMI-S, the values of interest are "GetStatisticsCollection", "Manifest Creation", "Manifest Modification", and "Manifest Removal".

41.3.7.3 AsynchronousMethodsSupported

This property within the OperationalPowerStatisticsCapabilities class defines the asynchronous mechanisms that are supported for retrieving statistics. For this release of SMI-S, this should be NULL.

41.3.7.4 ClockTickInterval

An internal clocking interval for all timer counters kept in the system implementation, measured in microseconds (i.e., the unit of measure in the timers, measured in microseconds). Time counters are considered to be monotonically increasing counters that contain "ticks". Each tick represents one clock tick interval.

For example, if ClockTickInterval contained a value of 32, then each time counter tick would represent 32 microseconds.

41.3.8 Health and Fault Management Consideration

Not defined in this version of the specification.

41.3.9 Cascading Considerations

Not applicable

41.4 Methods of the Profile

41.4.1 Extrinsic Methods of the Profile

41.4.1.1 Overview

The methods supported by this profile are summarized in Table 355 and detailed within the sections that follow it.

Table 355 - Creation, Deletion and Modification Methods

Method	Created Instances	Deleted Instances	Modified Instances
GetStatisticsCollection	None	None	None
CreateManifestCollection	OperationalPowerStatisticsManifestCollection AssociatedOperationalPowerStatisticsManifestCollection	None	None
AddOrModifyManifest	OperationalPowerStatisticsManifest(subclass) MemberOfCollection	None	OperationalPowerStatisticsManifest(subclass)
RemoveManifest	None	OperationalPowerStatisticsManifest(subclass) MemberOfCollection	None

41.4.1.2 GetStatisticsCollection

This extrinsic method retrieves statistics in a well-defined bulk format. The set of statistics returned by this method is determined by the list of element types passed into the method and the manifests for those

types contained in the supplied manifest collection. The statistics are returned through a well-defined array of strings that can be parsed to retrieve the desired statistics as well as limited information about the elements that those metrics describe.

```

GetStatisticsCollection(

    [OUT, Description("Reference to the job(shall be null in this version of SMI-S.)")
    CIM_ConcreteJob REF Job = NULL,

    [IN, Description("Element types for which statistics should be returned")
    ValueMap {"1", "2", "10", "201", "202", "203", "204", "205", "206", "207", "..",
              "0x8000.."},
    Values {"Other", "Top-Level System", "Disk Drive", "PDU", "Power Supply", "Fan",
            "CPU", "RAID Group", "Disk tray", "Controller", "DMTF
            Reserved", "Vendor Specific"}]

    uint16 ElementTypes[],

    [IN, Description ("An array of strings that specify the particular 'Other'
    element(s) when the ElementType property above includes the"
    "ElementType value of 1 (i.e., "Other"). Each string within"
    "this array identifies a separate "Other" element and duplicate"
    "string values are NOT allowed. This property should be set"
    "to NULL when the ElementType property does not include the" "value
    of 1.")]

    string OtherElementTypeDescriptions[],

    [IN, Description("The manifest collection that contains the manifests which list"
    "the metrics that should be returned for each element
    type")]

    CIM_OperationalPowerStatisticsManifestCollection REF ManifestCollection,

    [IN, Description("Specifies the format of the Statistics output parameter"),
    ValueMap {"2", "..", "0x8000.."},
    Values ("CSV", "DMTF Reserved", "Vendor Specific")]

    uint16 StatisticsFormat,

    [OUT, Description("The statistics for all the elements as determined by the"
    "Elements and ManifestCollection parameters")]

    string Statistics[]

);

```

Error returns are:

```

{"Not Supported", "Unknown", "Timeout", "Failed", "Invalid Parameter", "Method Reserved", "Method
Parameters", "Element Not Supported", "Statistics Format Not Supported", "Method Reserved", "Vendor
Specific"}

```

NOTE In this version of the standard, Job Control is not supported for the GetStatisticsCollection method. This method should always return NULL for the Job parameter, and no job related errors shall be returned.

If the ElementTypes[] array is empty, then no data shall be returned. If the ElementTypes[] array is NULL, then the ElementTypes[] parameter shall be ignored and all data specified in the manifest collection shall be returned.

If the manifest collection is empty, then no data shall be returned. If the manifest collection parameter is NULL, then the default manifest collection is used. (Note: In SMI-S, a default manifest collection shall exist if the `GetStatisticalCollection` method is supported).

NOTE The `ElementTypes[]` and `ManifestCollection` parameters may identify different sets of element types. The effect of this will be for the implementation to return statistics for the element types that are in both lists (that is, the intersection of the two lists). This intersection could be empty. In this case, no data will be returned.

For the current version of SMI-S, the only recognized value for `StatisticsFormat` is "CSV". The method may support other values, but they are not specified by SMI-S (i.e., they would be vendor specific).

Given a client has an inventory of the metered objects with `Statistics InstanceIDs` that may be used to correlate with the `OperationalPowerStatisticalData` instances, a simple CSV format is sufficient and the most efficient human-readable format for transferring bulk statistics. More specifically, the following rules constrain that format and define the content of the `String[]` `Statistics` output parameter to the `GetStatisticsCollection()` method:

- The `Statistics[]` array may contain multiple statistics records per array entry. In such cases, the total length of the concatenated record strings shall not exceed 64K bytes. And a single statistics record shall not span Array entries.
- There shall be exactly one statistics record per line in the bulk `Statistics` parameter. A line is terminated by:
 - a line-feed character
 - the end of a String Array Element (i.e., a statistics record cannot overlap elements of the `String[]` `Statistics` output parameter).
- Each statistics record shall contain the `InstanceID` of the `OperationalPowerStatisticalData` instance, the value map (number) of the `ElementType` of the metered object, and one value for each property that the relevant `OperationalPowerStatisticsManifest` specifies as "true".
- Each value in a record shall be separated from the next value by a Semi-colon (";"). This is to support internationalization of the CSV format. A provider creating a record in this format should not include white space between values in a record, though it may be included for human readability if so desired. A client reading a record it has received shall ignore white-space between values.
- The `InstanceID` value is an opaque string that shall correspond to the `InstanceID` property from `OperationalPowerStatisticalData` instance.
 - For the convenience of client software that needs to be able to correlate `InstanceIDs` between different `GetStatisticsCollection` method invocations, the `InstanceID` for `OperationalPowerStatisticalData` instance shall be unique across all instances of the `OperationalPowerStatisticalData` class. It is not sufficient that `InstanceID` is unique across subclasses of `OperationalPowerStatisticalData`.
- The `ElementType` value shall be a decimal string representation of the Element Type number (e.g., "201" for PDU). The `StatisticTime` shall be a string representation of `DateTime`. All other values shall be decimal string representations of their statistical values.
- Null values shall be included in records for which a statistic is returned (specified by the manifest or by a lack of manifest for a particular element type) but there is no meaningful value available for the statistic. A NULL statistic is represented by placing a semi-colon (;) in the record without a value at the position where the value would have otherwise been included. A record in which the last statistic has a NULL value shall end in a semi-colon (;). Clients shall ignore whitespace between semicolons.
- The first three values in a record shall be the `InstanceID`, `ElementType` and `StatisticTime` values from the `OperationalPowerStatisticalData` instance. The remaining values shall be returned in the order in which they are defined by the MOF for the `OperationalPowerStatisticsManifest` class or subclass the record describes.

As an additional convention, a provider should return all the records for a particular element type in consecutive String elements, and the order of the element types should be the same as the order in which the element types were specified in the input parameter to `GetStatisticsCollection()`.

41.4.1.3 CreateManifestCollection

This extrinsic method creates a new manifest collection whose members serve as a filter for metrics retrieved through the `GetStatisticsCollection` method.

```

CreateManifestCollection(
    [IN, Description("The collection of statistics that will be filtered using the new
                    manifest collection")]
    CIM_StatisticsCollection REF Statistics,

    [IN, Description("Client-defined name for the new manifest collection")]
    string ElementName,

    [OUT, Description("Reference to the new manifest collection")]
    CIM_OperationalPowerStatisticsManifestCollection REF ManifestCollection
);

```

Error returns are:

```

{"Ok", "Not Supported", "Unknown", "Timeout", "Failed", "Invalid Parameter",
 "Method Reserved", "Vendor Specific"}

```

41.4.1.4 AddOrModifyManifest

This is an extrinsic method that either creates or modifies a statistics manifest for this statistics service. A client supplies a manifest collection within which the new manifest collection will be placed or an existing manifest will be modified, the element type of the statistics that the manifest will filter, and a list of statistics that should be returned for that element type using the `GetStatisticsCollection` method.

```

AddOrModifyManifest(
    [IN, Description("Manifest collection that the manifest is or should be a member
                    of")]
    CIM_OperationalPowerStatisticsManifestCollection REF ManifestCollection,

    [IN, Description("The element type whose statistics the manifest will filter")
    ValueMap {"1", "2", "10", "201", "202", "203", "204", "205", "206", "207", "..",
             "0x8000.."},
    Values {"Other", "Disk Drive", "PDU", "Power Supply", "Fan", "CPU", "RAID Group",
            "Disk tray", "Controller", "DMTF Reserved", "Vendor
            Specific"}}]
    uint16 ElementType,

    [IN, Description ("A string describing the type of element when the ElementType
                    'property' above is set to 1 (i.e., 'Other'). This property
                    'should be set to NULL when the ElementType property is any"
                    "value other than 1.")]]
    string OtherElementTypeDescription,

    [IN, Description("The client-defined string that identifies the manifest created
                    or modified by this method")]

```

Operational Power Profile

```
string ElementName,  
  
[IN, Description("The statistics that will be included by the manifest filter;"  
                "that is, the statistics that will be supplied through"  
                "the" "GetStatisticsCollection method")  
string StatisticsList[],  
  
[OUT, Description("The Manifest that is created or modified on the successful"  
                 "execution of this method")]  
CIM_OperationalPowerStatisticsManifest REF Manifest  
);
```

Error returns are:

```
{"Success", "Not Supported", "Unknown", "Timeout", "Failed", "Invalid Parameter",  
 "Method Reserved", "Element Not Supported", "Metric not  
supported", "ElementType Parameter Missing", "Method  
Reserved", "Vendor Specific"}
```

If the StatisticsList[] array is empty, then only InstanceID and ElementType will be returned when the manifest is referenced. If the StatisticsList[] array parameter is NULL, then all supported properties is assumed (i.e., all supported properties shall be included).

NOTE This would be the OperationalPowerStatisticsManifest from the default manifest collection.

41.4.1.5 RemoveManifests

This is an extrinsic method that removes manifests from the manifest collection.

```
RemoveManifests(  
[IN, Description("Manifest collection from which the manifests will be removed")]  
CIM_OperationalPowerStatisticsManifestCollection REF ManifestCollection,  
  
[IN, Description("List of manifests to be removed from the manifest collection")  
CIM_OperationalPowerStatisticsManifest REF Manifest[]];
```

Error returns are:

```
{"Success", "Not Supported", "Unknown", "Timeout", "Failed", "Invalid Parameter",  
 "Method Reserved", "Manifest not found", "Method  
Reserved", "Vendor Specific"}
```

41.4.2 Intrinsic Methods of this Profile

41.4.2.1 DeleteInstance (of a OperationalPowerStatisticsManifestCollection)

This will delete the OperationalPowerStatisticsManifestCollection where IsDefault=False, the AssociatedOperationalPowerStatisticsManifestCollection association to the StatisticsCollection and all manifests collected by the manifest collection (and the MemberOfCollection associations to the OperationalPowerStatisticsManifestCollection).

41.4.2.2 Association Traversal

One of the ways of retrieving statistics is through association traversal from the StatisticsCollection to the individual Statistics following the MemberOfCollection association. This shall be supported by all

implementations of the Operational Power Profile and would be available to clients if the provider does not support the GetStatisticsCollection approach.

41.5 Use Cases

Not defined in this standard

41.6 Client Considerations and Recipes

41.6.1 Operations Using CIM_DeviceSet

The purpose of CIM_DeviceSet is to organize a set of LogicalDevices that support some management operation, for example the Operational Power management that instrumentation might provide to a client. The LogicalDevices associated to the CIM_DeviceSet would typically be related by some hardware constraint, like being in the same RAID Group that supports power operations.

41.6.2 Recipes

No recipes are defined in this version of the standard.

41.7 CIM Elements

Table 356 describes the CIM elements for Operational Power.

Table 356 - CIM Elements for Operational Power

Element Name	Requirement	Description
41.7.1 CIM_ElementCapabilities	Mandatory	This associates the OperationalPowerStatisticsCapabilities to the OperationalPowerStatisticsService.
41.7.2 CIM_ElementStatisticalData (Component System Stats)	Conditional	Conditional requirement: Component Systems statistics support. This is mandatory if OperationalPowerStatisticsCapabilities.ElementTypesSupported = '3', '4' or '5'. This associates an OperationalPowerStatisticalData instance to the component ComputerSystem for which the statistics are collected.
41.7.3 CIM_ElementStatisticalData (Top Level System Stats)	Conditional	Conditional requirement: Top level system statistics support. This is mandatory if OperationalPowerStatisticsCapabilities.ElementTypesSupported = '2'. This associates an OperationalPowerStatisticalData instance to the Top Level ComputerSystem for which the statistics are collected.
41.7.4 CIM_ElementStatisticalData (Volume Stats)	Optional	This is mandatory if OperationalPowerStatisticsCapabilities.ElementTypesSupported = '8', and the parent profile supports Storage Volumes. This associates an OperationalPowerStatisticalData instance to the volume for which the statistics are collected.

Table 356 - CIM Elements for Operational Power

Element Name	Requirement	Description
41.7.5 CIM_HostedCollection (Client Defined)	Conditional	Conditional requirement: Clients can create manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or Clients can create manifests as identified by OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported. This would associate a client defined OperationalPowerStatisticalManifestCollection to the top level system for the profile (e.g., array).
41.7.6 CIM_HostedCollection (Default)	Mandatory	This would associate a default OperationalPowerStatisticsManifestCollection to the top level system for the profile (e.g., array).
41.7.7 CIM_HostedCollection (System to StatisticsCollection)	Mandatory	This would associate the StatisticsCollection to the top level system for the profile (e.g., array).
41.7.8 CIM_HostedService	Mandatory	This associates the OperationalPowerStatisticsService to the ComputerSystem that hosts it.
41.7.9 CIM_MemberOfCollection (DeviceSet)	Optional	This would associate LogicalDevices to CIM_DeviceSets.
41.7.10 CIM_MemberOfCollection (Member of client defined collection)	Conditional	Conditional requirement: Clients can modify manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported. This would associate Manifests to client defined manifest collections.
41.7.11 CIM_MemberOfCollection (Member of pre-defined collection)	Mandatory	This would associate pre-defined Manifests to default manifest collection.
41.7.12 CIM_MemberOfCollection (Member of statistics collection)	Mandatory	This would associate all statistics instances to the StatisticsCollection.
41.7.13 CIM_StatisticsCollection	Mandatory	This would be a collection point for all Statistics that are kept for a storage system.
41.7.14 CIM_DeviceSet (Provider Defined)	Optional	An instance of this class defines a grouping of LogicalDevices that support some management operation.
41.7.15 CIM_OperationalPowerManifest (Client Defined)	Conditional	Conditional requirement: Clients can modify manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported. An instance of this class defines the statistics properties of interest to the client for one element type.
41.7.16 CIM_OperationalPowerManifest (Provider Support)	Mandatory	An instance of this class defines the statistics properties supported by the profile implementation for one element type.
41.7.17 CIM_OperationalPowerManifestCollection (Client Defined)	Conditional	Conditional requirement: Clients can create manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported. An instance of this class defines one client defined collection of statistics manifests (one manifest for each element type).
41.7.18 CIM_OperationalPowerManifestCollection (Provider Defined)	Mandatory	An instance of this class defines the predefined collection of default statistics manifests (one manifest for each element type).

Table 356 - CIM Elements for Operational Power

Element Name	Requirement	Description
41.7.19 CIM_OperationalPowerStatisticalData	Mandatory	This is a Subclass of CIM_StatisticalData for Operational Power statistics. It is instantiated to provide specific statistics for particular components.
41.7.20 CIM_OperationalPowerStatisticsCapabilities	Mandatory	This defines the statistics capabilities supported by the implementation of the profile.
41.7.21 CIM_OperationalPowerStatisticsService	Mandatory	This is a Service that provides (optional) services of bulk statistics retrieval and manifest set manipulation methods.

41.7.1 CIM_ElementCapabilities

CIM_ElementCapabilities represents the association between ManagedElements (i.e., OperationalPowerStatisticsService) and their Capabilities (e.g., OperationalPowerStatisticsCapabilities). Note that the cardinality of the ManagedElement reference is Min(1), Max(1). This cardinality mandates the instantiation of the CIM_ElementCapabilities association for the referenced instance of Capabilities. ElementCapabilities describes the existence requirements and context for the referenced instance of ManagedElement. Specifically, the ManagedElement shall exist and provides the context for the Capabilities.

CIM_ElementCapabilities is not subclassed from anything.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 357 describes class CIM_ElementCapabilities.

Table 357 - SMI Referenced Properties/Methods for CIM_ElementCapabilities

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	The managed element (OperationalPowerStatisticsService).
Capabilities		Mandatory	The Capabilities instance associated with the OperationalPowerStatisticsService.

41.7.2 CIM_ElementStatisticalData (Component System Stats)

CIM_ElementStatisticalData is an association that relates a component ComputerSystem to its statistics. Note that the cardinality of the ManagedElement reference is Min(1), Max(1). This cardinality mandates the instantiation of the CIM_ElementStatisticalData association for the referenced instance of OperationalPowerStatisticalData. ElementStatisticalData describes the existence requirements and context for the OperationalPowerStatisticalData, relative to a specific component ComputerSystem.

CIM_ElementStatisticalData is not subclassed from anything.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Component Systems statistics support.

Table 358 describes class CIM_ElementStatisticalData (Component System Stats).

Table 358 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Component System Stats)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	A reference to a component ComputerSystem for which the Statistics apply.
Stats		Mandatory	A reference to the OperationalPowerStatisticalData that hold the statistics for the ComputerSystem.

41.7.3 CIM_ElementStatisticalData (Top Level System Stats)

CIM_ElementStatisticalData is an association that relates a top level ComputerSystem to its statistics. Note that the cardinality of the ManagedElement reference is Min(1), Max(1). This cardinality mandates the instantiation of the CIM_ElementStatisticalData association for the referenced instance of OperationalPowerStatisticalData. ElementStatisticalData describes the existence requirements and context for the OperationalPowerStatisticalData, relative to a specific ComputerSystem.

CIM_ElementStatisticalData is not subclassed from anything.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Top level system statistics support.

Table 359 describes class CIM_ElementStatisticalData (Top Level System Stats).

Table 359 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Top Level System Stats)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	A reference to the top level ComputerSystem for which the Statistics apply.
Stats		Mandatory	A reference to the OperationalPowerStatisticalData that hold the statistics for the ComputerSystem.

41.7.4 CIM_ElementStatisticalData (Volume Stats)

CIM_ElementStatisticalData is an association that relates a StorageVolume to its statistics. Note that the cardinality of the ManagedElement reference is Min(1), Max(1). This cardinality mandates the instantiation of the CIM_ElementStatisticalData association for the referenced instance of OperationalPowerStatisticalData. ElementStatisticalData describes the existence requirements and context for the OperationalPowerStatisticalData, relative to a specific volume.

CIM_ElementStatisticalData is not subclassed from anything.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 360 describes class CIM_ElementStatisticalData (Volume Stats).

Table 360 - SMI Referenced Properties/Methods for CIM_ElementStatisticalData (Volume Stats)

Properties	Flags	Requirement	Description & Notes
ManagedElement		Mandatory	A reference to a StorageVolume for which the Statistics apply.
Stats		Mandatory	A reference to the OperationalPowerStatisticalData that hold the statistics for the StorageVolume.

41.7.5 CIM_HostedCollection (Client Defined)

CIM_HostedCollection defines a SystemSpecificCollection in the context of a scoping System. It represents a Collection that only has meaning in the context of a System, and/or whose elements are restricted by the definition of the System. In the Operational Power profile, it is used to associate a client defined OperationalPowerStatisticalManifestCollections to the top level Computer System.

CIM_HostedCollection is subclassed from CIM_HostedDependency.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Clients can create manifests as identified by

OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or Clients can create manifests as identified by OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported.

Table 361 describes class CIM_HostedCollection (Client Defined).

Table 361 - SMI Referenced Properties/Methods for CIM_HostedCollection (Client Defined)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	The top level ComputerSystem of the profile.
Dependent		Mandatory	A client defined OperationalPowerStatisticalManifestCollection.

41.7.6 CIM_HostedCollection (Default)

CIM_HostedCollection defines a SystemSpecificCollection in the context of a scoping System. It represents a Collection that only has meaning in the context of a System, and/or whose elements are restricted by the definition of the System. In the Operational Power profile, it is used to associate the default OperationalPowerStatisticsManifestCollection to the top level Computer System.

CIM_HostedCollection is subclassed from CIM_HostedDependency.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 362 describes class CIM_HostedCollection (Default).

Table 362 - SMI Referenced Properties/Methods for CIM_HostedCollection (Default)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	The top level ComputerSystem of the profile.
Dependent		Mandatory	The provider defined OperatonaPowerManifestCollection.

41.7.7 CIM_HostedCollection (Systemto StatisticsCollection)

CIM_HostedCollection defines a SystemSpecificCollection in the context of a scoping System. It represents a Collection that only has meaning in the context of a System, and/or whose elements are restricted by the definition of the System. In the Operational Power profile, it is used to associate the StatisticsCollection to the top level Computer System.

CIM_HostedCollection is subclassed from CIM_HostedDependency.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 363 describes class CIM_HostedCollection (Systemto StatisticsCollection).

Table 363 - SMI Referenced Properties/Methods for CIM_HostedCollection (Systemto StatisticsCollection)

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	The top level ComputerSystem of the profile.
Dependent		Mandatory	The StatisticsCollection.

41.7.8 CIM_HostedService

CIM_HostedService is an association between a Service (OperationalPowerStatisticsService) and the System (ComputerSystem) on which the functionality resides. Services are weak with respect to their hosting System. Heuristic: A Service is hosted on the System where the LogicalDevices or SoftwareFeatures that implement the Service are located.

CIM_HostedService is subclassed from CIM_HostedDependency.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 364 describes class CIM_HostedService.

Table 364 - SMI Referenced Properties/Methods for CIM_HostedService

Properties	Flags	Requirement	Description & Notes
Antecedent		Mandatory	The hosting System.
Dependent		Mandatory	The Service hosted on the System.

41.7.9 CIM_MemberOfCollection (DeviceSet)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 365 describes class CIM_MemberOfCollection (DeviceSet).

Table 365 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (DeviceSet)

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	The CIM_DeviceSet.
Member		Mandatory	The individual LogicalDevice that is part of the set.

41.7.10 CIM_MemberOfCollection (Member of client defined collection)

This use of MemberOfCollection is to Collect all Manifests instances in a client defined manifest collection.

Created By: Extrinsic: AddOrModifyManifest

Modified By: Static

Deleted By: Extrinsic: RemoveManifest

Requirement: Clients can modify manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported.

Table 366 describes class CIM_MemberOfCollection (Member of client defined collection).

Table 366 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Member of client defined collection)

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	A client defined manifest collection.
Member		Mandatory	The individual Manifest Instance that is part of the set.

41.7.11 CIM_MemberOfCollection (Member of pre-defined collection)

This use of MemberOfCollection is to Collect all Manifests instances in the default manifest collection.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 367 describes class CIM_MemberOfCollection (Member of pre-defined collection).

Table 367 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Member of pre-defined collection)

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	The provider defined default manifest collection.
Member		Mandatory	The individual Manifest Instance that is part of the set.

41.7.12 CIM_MemberOfCollection (Member of statistics collection)

This use of MemberOfCollection is to collect all OperationalPowerStatisticalData instances (in the StatisticsCollection). Each association is created as a side effect of the metered object getting created.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 368 describes class CIM_MemberOfCollection (Member of statistics collection).

Table 368 - SMI Referenced Properties/Methods for CIM_MemberOfCollection (Member of statistics collection)

Properties	Flags	Requirement	Description & Notes
Collection		Mandatory	The default manifest collection.
Member		Mandatory	The individual Manifest Instance that is part of the set.

41.7.13 CIM_StatisticsCollection

The CIM_StatisticsCollection collects all statistics kept by the profile. There is one instance of the CIM_StatisticsCollection class and all individual element statistics can be accessed by using association traversal(using MemberOfCollection) from the StatisticsCollection.

CIM_StatisticsCollection is subclassed from CIM_SystemSpecificCollection.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 369 describes class CIM_StatisticsCollection.

Table 369 - SMI Referenced Properties/Methods for CIM_StatisticsCollection

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
SampleInterval		Mandatory	Minimum recommended polling interval for an array, storage virtualizer system or volume manager. It is set by the provider and cannot be modified.
TimeLastSampled		Mandatory	Time statistics table by object was last updated (Time Stamp in SMI 2.2 specification format).

41.7.14 CIM_DeviceSet (Provider Defined)

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Optional

Table 370 describes class CIM_DeviceSet (Provider Defined).

Table 370 - SMI Referenced Properties/Methods for CIM_DeviceSet (Provider Defined)

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	Unique identifier for the collection.
ElementName		Mandatory	User friendly name for the collection.
Purpose		Mandatory	This property shows what kind of management operation is the purpose of the grouping.
GroupType		Mandatory	This property shows the unit of grouping.

41.7.15 CIM_OperationalPowerManifest (Client Defined)

The CIM_OperationalPowerManifest class is concrete class that defines the OperationalPowerStatisticalData properties that should be returned on a GetStatisticsCollection request.

CIM_OperationalPowerManifest is subclassed from CIM_ManagedElement.

In order for a client defined instance of the CIM_OperationalPowerManifest class to exist, the all the manifest collection manipulation functions shall be identified in the 'SynchronousMethodsSupported' property of the OperationalPowerStatisticsCapabilities (OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported = '6') instance, AND a client must have created at least ONE instance of OperationalPowerManifestCollection.

Created By: Extrinsic: AddOrModifyManifest

Modified By: Extrinsic: AddOrModifyManifest

Deleted By: Extrinsic: RemoveManifest

Requirement: Clients can modify manifests as identified by CIM_OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported.

Table 371 describes class CIM_OperationalPowerManifest (Client Defined).

Table 371 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifest (Client Defined)

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	A Client defined string that identifies the manifest.
InstanceID		Mandatory	The instance Identification. Within the scope of the instantiating Namespace, InstanceID opaquely and uniquely identifies an instance of this class.
ElementType		Mandatory	This value is required AND the current version of SMI-S specifies the following values: ValueMap {'2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12'} Values { 'Computer System', 'Front-end Computer System', 'Peer Computer System', 'Back-end Computer System', 'Front-end Port', 'Back-end Port', 'Volume', 'Extent', 'Disk Drive', 'Arbitrary LUs', 'Remote Replica Group'}.

Table 371 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifest (Client Defined)

Properties	Flags	Requirement	Description & Notes
IncludeStatisticTime		Mandatory	
CSVSequence		Mandatory	An array of strings that define a sequence of OperationalPowerStatisticalData property names. The sequence is the sequence that data is to be returned on a GetStatisticsCollection request using this manifest. The first three elements of this array should be "InstanceID", "ElementType" and "StatisticsTime" to allow applications to match the ElementType of the Manifest with the OperationalPowerStatisticalData CSV record. For OperationalPowerManifest (Client Defined) this shall be the sequence desired by the client.

41.7.16 CIM_OperationalPowerManifest (Provider Support)

The CIM_OperationalPowerManifest class is concrete class that defines the OperationalPowerStatisticalData properties that supported by the Provider. These Manifests are established by the Provider for the default manifest collection.

CIM_OperationalPowerManifest is subclassed from CIM_ManagedElement.

At least one Provider supplied instance of the CIM_OperationalPowerManifest class shall exist, if the Operational Power profile is supported.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 372 describes class CIM_OperationalPowerManifest (Provider Support).

Table 372 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifest (Provider Support)

Properties	Flags	Requirement	Description & Notes
ElementName		Mandatory	A Provider defined string that identifies the manifest in the context of the Default Manifest Collection.
InstanceID		Mandatory	The instance Identification. Within the scope of the instantiating Namespace, InstanceID opaquely and uniquely identifies an instance of this class.
ElementType		Mandatory	This value is required AND the current version of SMI-S specifies the following values: ValueMap {'2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12'} Values {'Computer System', 'Front-end Computer System', 'Peer Computer System', 'Back-end Computer System', 'Front-end Port', 'Back-end Port', 'Volume', 'Extent', 'Disk Drive', 'Arbitrary LUs', 'Remote Replica Group'}.

Table 372 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifest (Provider Support)

Properties	Flags	Requirement	Description & Notes
IncludeStatisticTime		Mandatory	
CSVSequence		Mandatory	An array of strings that define a sequence of OperationalPowerStatisticalData property names. The sequence is the sequence that data is to be returned on a GetStatisticsCollection request using this manifest. The first three elements of this array shall be "InstanceID", "ElementType" and "StatisticsTime" to allow applications to match the ElementType of the Manifest with the OperationalPowerStatisticalData CSV record. For OperationalPowerManifest (Provider Support) this shall be the default sequence provided by the provider.

41.7.17 CIM_OperationalPowerManifestCollection (Client Defined)

An instance of a client defined CIM_OperationalPowerManifestCollection defines the set of Manifests to be used in retrieval of statistics by the GetStatisticsCollection method.

CIM_OperationalPowerManifestCollection is subclassed from CIM_SystemSpecificCollection.

In order for a client defined instance of the CIM_OperationalPowerManifestCollection class to exist, then all the manifest collection manipulation functions shall be identified in the 'SynchronousMethodsSupported' property of the CIM_OperationalPowerStatisticsCapabilities instance and a client must have created a Manifest Collection..

Created By: Extrinsic: CreateManifestCollection

Modified By: Static

Deleted By: Static

Requirement: Clients can create manifests as identified by CIM_OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported.

Table 373 describes class CIM_OperationalPowerManifestCollection (Client Defined).

Table 373 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifestCollection (Client Defined)

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	A client defined user-friendly name for the manifest collection. It is set during creation of the Manifest Collection through the ElementName parameter of the CreateManifestCollection method.
IsDefault		Mandatory	Denotes whether or not this manifest collection is a provider defined default manifest collection. For the client defined manifest collections this is set to 'false'.

41.7.18 CIM_OperationalPowerManifestCollection (Provider Defined)

An instance of a default OperationalPowerManifestCollection defines the set of Manifests that define the properties supported for each ElementType supported for the implementation. It can also be used by clients in retrieval of statistics by the GetStatisticsCollection method.

CIM_OperationalPowerManifestCollection is subclassed from CIM_SystemSpecificCollection.

At least ONE CIM_OperationalPowerManifestCollection shall exist if the Operational Power profile is implemented. This would be the default manifest collection that defines the properties supported by the implementation.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 374 describes class CIM_OperationalPowerManifestCollection (Provider Defined).

Table 374 - SMI Referenced Properties/Methods for CIM_OperationalPowerManifestCollection (Provider Defined)

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	For the default manifest collection, this should be set to 'DEFAULT'.
IsDefault		Mandatory	Denotes whether or not this manifest collection is a provider defined default manifest collection. For the default manifest collection this is set to 'true'.

41.7.19 CIM_OperationalPowerStatisticalData

The OperationalPowerStatisticalData class defines the statistics properties that may be kept for an metered element of the storage entity.

Instances of this class will exist for each of the metered elements if the 'ElementTypesSupported' property of the OperationalPowerStatisticsCapabilities indicates that the metered element is supported. For example, 'Computer System' is identified in the 'ElementTypesSupported' property, then this indicates support for metering of the Top level computer system or 'Component Computer System'.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 375 describes class CIM_OperationalPowerStatisticalData.

Table 375 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticalData

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	The InstanceID for OperationalPowerStatisticalData instance shall be unique across all instances of the OperationalPowerStatisticalData class.
StatisticTime		Mandatory	Time statistics table by object was last updated (Time Stamp in CIM 2.2 specification format).

Table 375 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticalData

Properties	Flags	Requirement	Description & Notes
ElementType		Mandatory	This value is required AND current version of SMI-S specifies the following values: ValueMap { '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12' } Values { 'Computer System', 'Front-end Computer System', 'Peer Computer System', 'Back-end Computer System', 'Front-end Port', 'Back-end Port', 'Volume', 'Extent', 'Disk Drive', 'Arbitrary LUs', 'Remote Replica Group' }.
Milliwatts		Mandatory	

41.7.20CIM_OperationalPowerStatisticsCapabilities

An instance of the OperationalPowerStatisticsCapabilities class defines the specific support provided with the statistics implementation. Note: There would be zero or one instance of this class in a profile. There would be none if the profile did not support the Operational Power profile. There would be exactly one instance if the profile did support the Operational Power Statistics profile.

CIM_OperationalPowerStatisticsCapabilities class is subclassed from CIM_Capabilities.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 376 describes class CIM_OperationalPowerStatisticsCapabilities.

Table 376 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticsCapabilities

Properties	Flags	Requirement	Description & Notes
InstanceID		Mandatory	
ElementName		Mandatory	
ElementTypesSupported		Mandatory	ValueMap { '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12' }, Values { 'Computer System', 'Front-end Computer System', 'Peer Computer System', 'Back-end Computer System', 'Front-end Port', 'Back-endPort', 'Volume', 'Extent', 'Disk Drive', 'Arbitrary LUs', 'Remote Replica Group' }.
SynchronousMethodsSupported	N	Mandatory	This property is mandatory, but the array may be empty. ValueMap { '2', '3', '4', '5', '6', '7', '8' }, Values { 'Exec Query', 'Indications', 'QueryCollection', 'GetStatisticsCollection', 'Manifest Creation', 'Manifest Modification', 'Manifest Removal' }.
AsynchronousMethodsSupported		Optional	Not supported in current version of SMI-S.
ClockTickInterval		Mandatory	An internal clocking interval for all timers in the subsystem, measured in microseconds (Unit of measure in the timers, measured in microseconds). Time counters are monotonically increasing counters that contain 'ticks'. Each tick represents one ClockTickInterval. If ClockTickInterval contained a value of 32 then each time counter tick would represent 32 microseconds.

41.7.21 CIM_OperationalPowerStatisticsService

The OperationalPowerStatisticsService class provides methods for statistics retrieval and Manifest Collection manipulation.

There shall be an instance of the OperationalPowerStatisticsService, if the Operational Power profile is implemented. It is not necessary to support any methods of the service, but the service shall be populated.

The methods that are supported can be determined from the SynchronousMethodsSupported and AsynchronousMethodsSupported properties of the OperationalPowerStatisticsCapabilities.

Created By: Static

Modified By: Static

Deleted By: Static

Requirement: Mandatory

Table 377 describes class CIM_OperationalPowerStatisticsService.

Table 377 - SMI Referenced Properties/Methods for CIM_OperationalPowerStatisticsService

Properties	Flags	Requirement	Description & Notes
SystemCreationClassName		Mandatory	
SystemName		Mandatory	
CreationClassName		Mandatory	
Name		Mandatory	
GetStatisticsCollection()		Conditional	Conditional requirement: Clients can get statistics collections using the GetStatisticsCollection as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or Clients can get statistics collections using the GetStatisticsCollection as identified by OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported. Support for this method is conditional on OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported containing '5' (GetStatisticsCollection). This method retrieves all statistics kept for the profile as directed by a manifest collection.
CreateManifestCollection()		Conditional	Conditional requirement: Clients can create manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or Clients can create manifests as identified by OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported. Support for this method is conditional on OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported containing '6' (Manifest Creation). This method is used to create client defined manifest collections.
AddOrModifyManifest()		Conditional	Conditional requirement: Clients can modify manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or Clients can modify manifests as identified by OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported. Support for this method is conditional on OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported containing '7' (Manifest Modification). This method is used to add or modify statistics manifests in a client defined manifest collection.
RemoveManifests()		Conditional	Conditional requirement: Clients can remove manifests as identified by OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or Clients can remove manifests as identified by OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported. Support for this method is conditional on OperationalPowerStatisticsCapabilities.SynchronousMethodsSupported or OperationalPowerStatisticsCapabilities.AsynchronousMethodsSupported containing '8' (Manifest Removal). This method is used to remove a statistics manifest from a client defined manifest collection.

EXPERIMENTAL

Operational Power Profile

42 Indications Profile

The Indications Profile is defined by the DMTF in DSP1054 Indications Profile.

DEPRECATED

The SNIA specialization of the DMTF Indications Profile has been removed in this version of the standard as the experimental specializations are no longer used. All implementers should refer to the DMTF Profile.

DEPRECATED

Indications Profile

EXPERIMENTAL

43 WBEM Server Profile

The WBEM Server profile is defined by the DMTF in DSP1092 WBEM Server Profile.

The WBEM Server profile is mandatory for all compliant SMI-S implementations

EXPERIMENTAL

WBEM Server Profile

Annex A (informative) SMI-S Information Model

This standard is based on DMTF's CIM schema, version 2.45.0. The DMTF schema is available in the machine-readable Managed Object Format (MOF) format. DMTF MOFs are simultaneously released both as an "Experimental" and a "Final" version of the schema. This provides developers with early access to experimental parts of the models. Both versions are available at

<http://www.dmtf.org/standards/cim>

Content marked as "Experimental" or "Implemented" may be based on DMTF's Experimental MOFs.

Annex A (informative) Cross Profile Considerations

A.1 Overview

Many applications access data from multiple profiles to perform operations. Figure B.1 shows a client application communicating with multiple SMI-S agents.

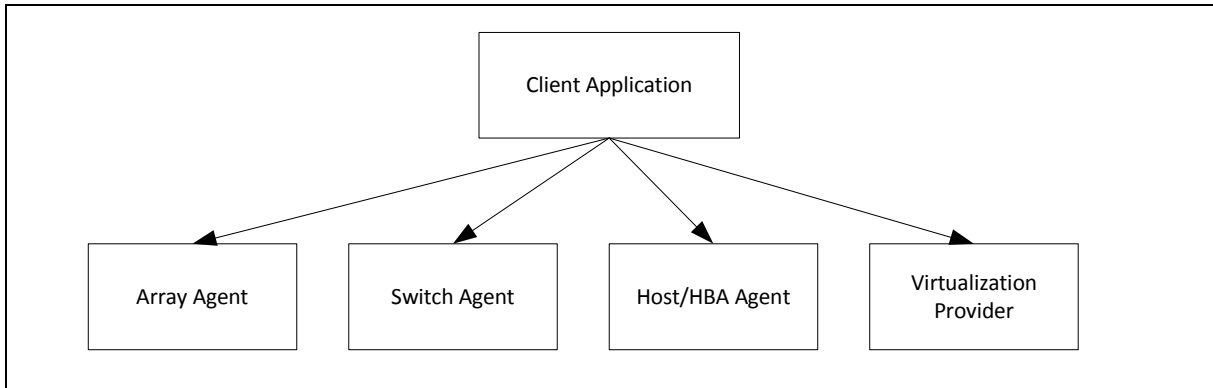


Figure B.1 - System Diagram

This section describes algorithms that can be used to associate objects from different profiles to understand connections between the profiles. The algorithms use Durable Names to match objects from different profiles. Figure B.2: "Host Bus Adapter Model" and Figure B.3: "Switch Model" are simplified instance diagrams that are used to illustrate the algorithms.

A.2 HBA model

Figure B.2: "Host Bus Adapter Model" represents a simple "Host Bus Adapter". The model includes objects that represent a single port Fibre channel HBA. The model also includes a storage volume being accessed through the HBA.

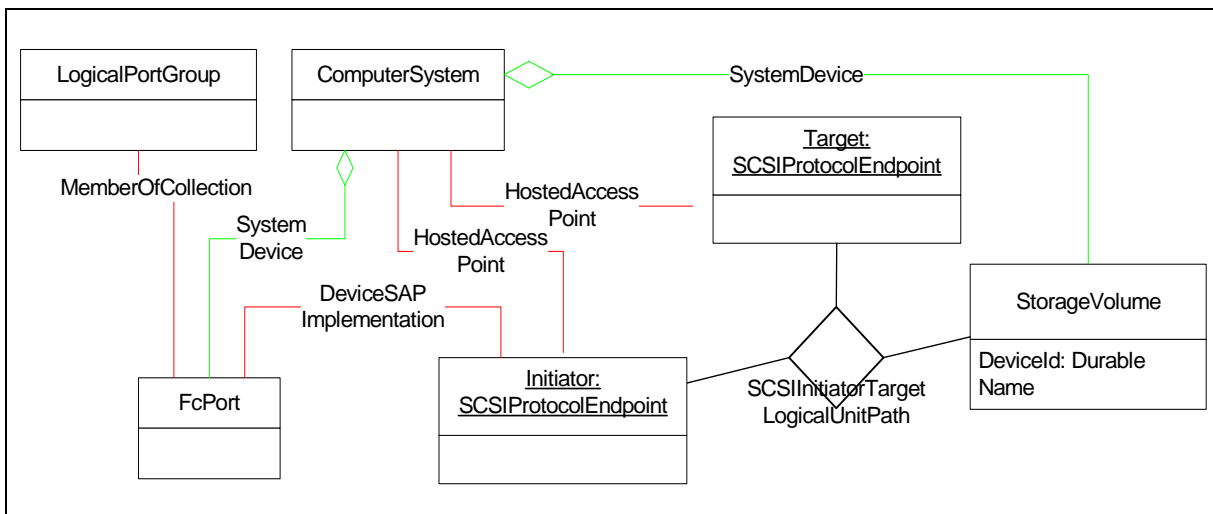


Figure B.2 - Host Bus Adapter Model

A.3 Switch Model

Figure B.3: "Switch Model" represents a two-port Fibre channel switch. The model also includes objects representing links to remote ports the switch agent knows about, and ComputerSystems.

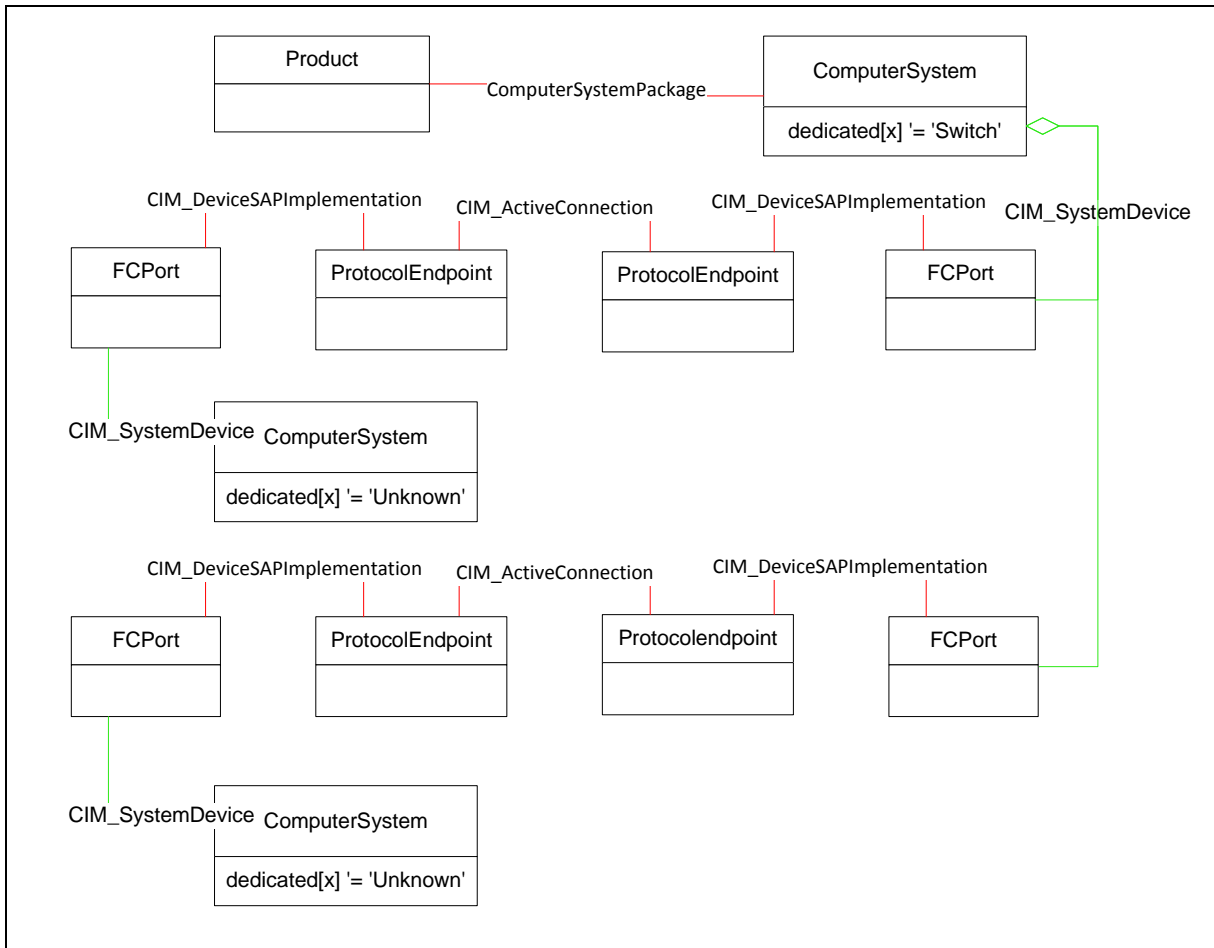


Figure B.3 - Switch Model

A.4 Array Model

Figure B.4: "Array Instance" is a simple model of a disk array. The array has a single controller with a single Fibre channel port on the front end and a single parallel SCSI port for the disks. The model shows two disks that are members of a single redundancy group. Part of the redundancy group is made available over the Fibre channel as a single volume.

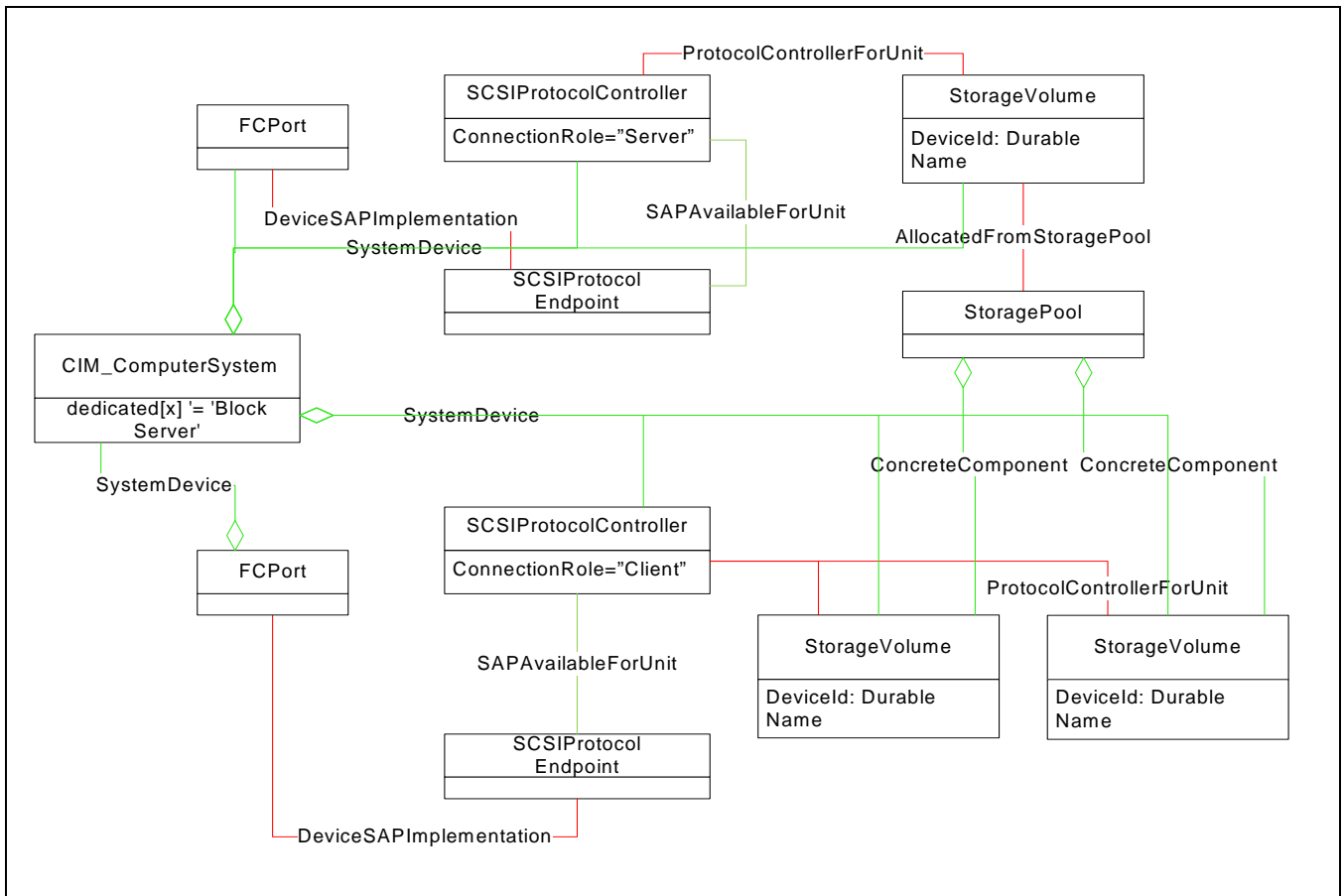


Figure B.4 - Array Instance

A.5 Storage Virtualization Model

Figure B.5: "Virtualization Instance" is a simple model of a Storage Virtualizer. The model shows the basic controller and pool. The model also shows a single volume being used and a single volume being served to a host.

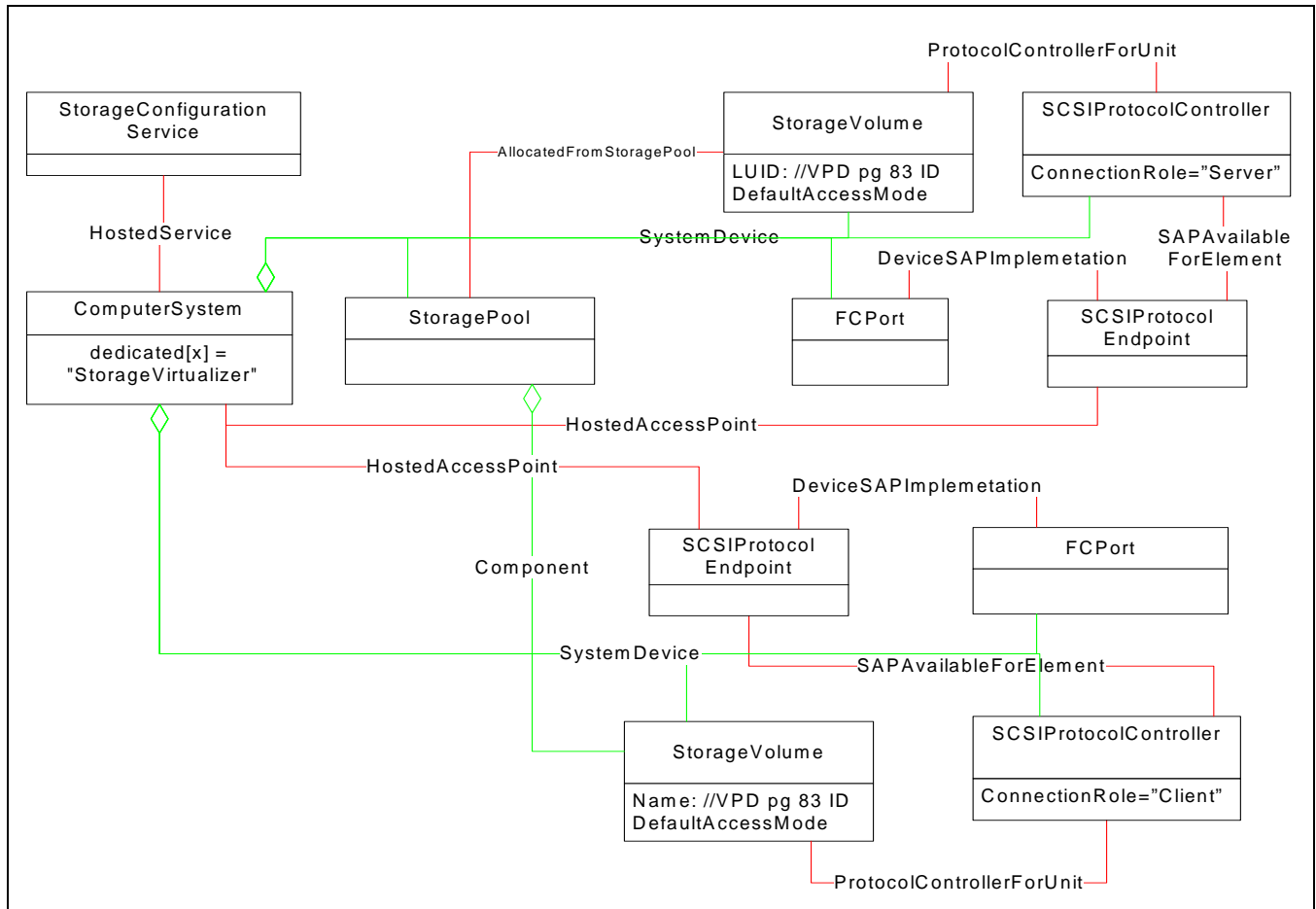


Figure B.5 - Virtualization Instance

A.6 Fabric Topology (HBA, Switch, Array)

A map of a SAN that shows all the elements and the connections between them is very useful. To create the map all the elements in the SAN with their Fibre channel ports are first located. Next the ports are linked together. Figure B.6 shows how a SAN map can be constructed.

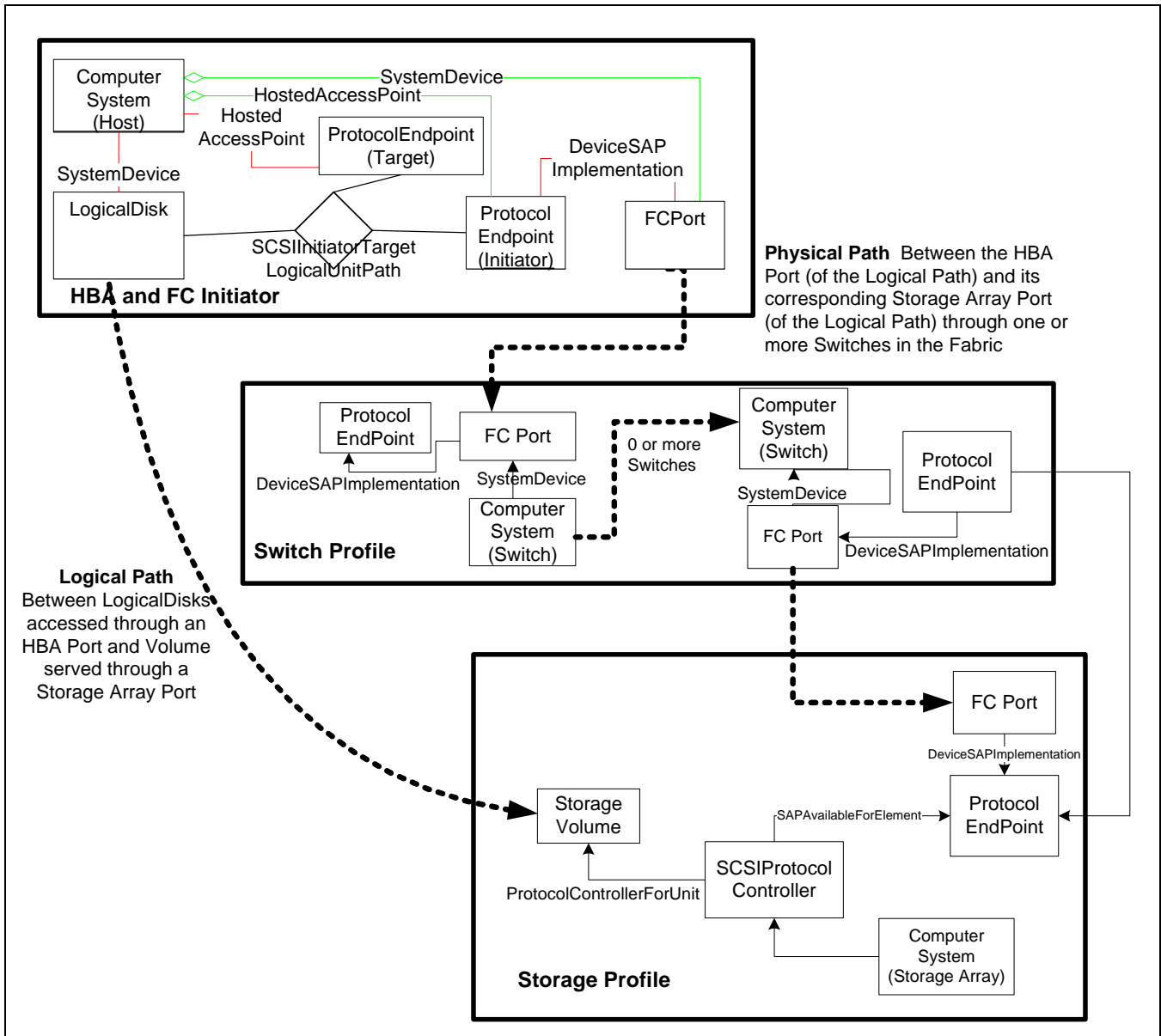


Figure B.6 - Fabric Topology

To locate all the elements in a SAN, you start by locating the agents. SMI-S agents are located using SLP. Once the agents are located, intrinsic methods are used to enumerate ComputerSystem objects. Each ComputerSystem object represents an element in the SAN. The ComputerSystem object's "Dedicated" attribute can be used to identify the type of the element.

After the elements are located, Fibre channel ports for each element are discovered. For each ComputerSystem object follow SystemDeviceFCPort objects. Use the information in the FCPort objects found to determine the Durable Name for the FCPort object. The Durable Name is used to match the ports to objects in other profiles.

Now to link the elements' ports together find the Switch elements. Switches know about ports on elements logged into their ports. To find this information start by locating the ComputerSystem objects that represents switches. Switches can be identified by the "Dedicated" attribute of the ComputerSystem object being set to "Switch". For each switch follow the SystemDeviceFCPort objects that represent the ports of the switch. Next look for ActiveConnection ActiveConnectionFCPort objects. These FCPort objects represent the ports on the other side of a link. Use attributes from the FCPort object to determine the Durable Name. These identifiers are then matched to identifiers found in other profiles to complete the connection.