



Storage Management Technical Specification, Part 2 Common Architecture

Version 1.8.0, Revision 3

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

3 November 2018

USAGE

Copyright © 2018 SNIA. All rights reserved. All other trademarks or registered trademarks are the property of their respective owners.

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

- 1) Any text, diagram, chart, table or definition reproduced 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 or any portion thereof, or distribute this document to third parties. All rights not explicitly granted are expressly reserved to SNIA.

Permission to use this document for purposes other than those enumerated above may be requested by 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) 2018, 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 © 2003-2018 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.

REVISION HISTORY

Revision 1

Date

April 14, 2017

SCRs Incorporated and other changes

Annex A (informative) SMI-S Information Model
- This standard is now based on DMTF's CIM schema Version 2.50.

Comments

Editorial notes and DRAFT material are displayed.

Revision 2

Date

February 16, 2018

SCRs Incorporated and other changes

None.

Comments

Substantial editorial clean up to align CIM references with existing XML.

Revision 3

Date

August 14, 2018

SCRs Incorporated and other changes

None.

Comments

None.

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

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 1.8.0. 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 document unless its initial architecture has been completed and reviewed. Some content included in this document 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.

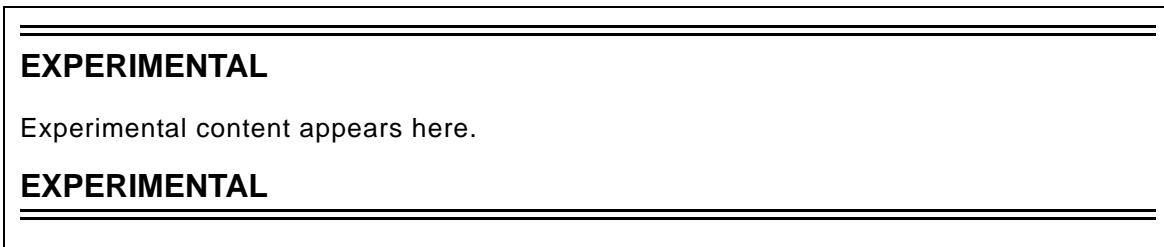


Figure 1 - Experimental Maturity Level Tag

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.

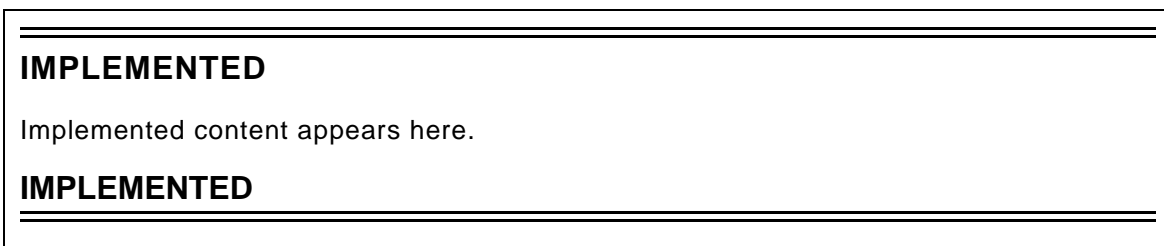


Figure 2 - Implemented Maturity Level Tag

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.

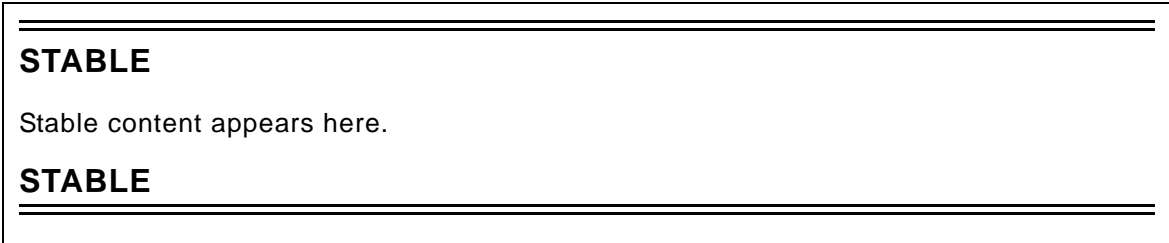


Figure 3 - Stable Maturity Level Tag

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.

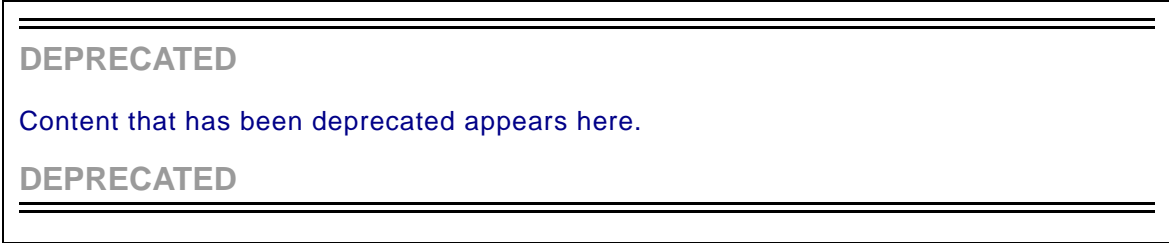


Figure 4 - Deprecated Tag

Contents

Revision History	5
List of Figures	15
List of Tables	17
Foreword	23
1 Scope	25
2 Normative references	27
2.1 General	27
2.2 Approved references	27
2.3 DMTF references (Final)	27
2.4 IETF references	28
2.5 References under development	29
2.6 Other references	29
3 Terms, definitions, symbols, abbreviations, and conventions	31
3.1 Terms and definitions	31
3.2 Acronyms and abbreviations	37
3.3 Keywords	37
3.4 Conventions	38
4 Transport and Reference Model	41
4.1 Introduction	41
4.1.1 Overview	41
4.1.2 Language Requirements	41
4.1.3 Communications Requirements	41
4.1.4 XML Message Syntax and Semantics	41
4.2 Transport Stack	42
4.3 Reference Model	42
4.3.1 Overview	42
4.3.2 Roles for Interface Constituents	43
4.3.3 Cascaded Agents	43
5 Health and Fault Management	45
5.1 Objectives	45
5.2 Overview	45
5.3 Terms	45
5.4 Description of Health and Fault Management	46
5.4.1 Operational Status and Health State (Polling)	46
5.4.2 Standard Errors and Events	47
5.4.3 Indications	47
5.4.4 Event Correlation and Fault Containment	47
5.4.5 Fault Regions	50
5.4.6 Examples	52
6 Object Model General Information	55
6.1 Model Overview (Key Resources)	55
6.1.1 Overview	55
6.1.2 Introduction to CIM UML Notation	55
6.2 Techniques	56
6.2.1 CIM Fundamentals	56
6.2.2 Modeling Profiles	58
6.2.3 CIM Naming	58
7 Correlatable and Durable Names	59

7.1	Overview	59
7.2	Guidelines for SCSI Logical Unit Names	60
7.3	Guidelines for FC-SB-2 Device Names.....	60
7.4	Guidelines for Port Names	61
7.5	Guidelines for Storage System Names	61
7.6	Standard Formats for Correlatable Names	62
7.6.1	General.....	62
7.6.2	Standard Formats for Logical Unit Names	63
7.6.3	Standard Formats for Port Names.....	64
7.6.4	Standard Formats for Fabric Names	65
7.6.5	Standard Formats for Storage System Names.....	65
7.6.6	Operating System Device Names	67
7.6.7	Case Sensitivity	68
7.7	Testing Equality of correlatable Names	68
7.8	iSCSI Names.....	69
8	Standard Messages.....	71
8.1	Overview	71
8.2	Registries for Standard Messages	71
8.3	SNIA Standard Messages.....	71
8.3.1	Common Profile-related Messages	71
8.3.2	Block Storage Messages.....	85
8.3.3	Fabric Messages	118
8.3.4	Filesystem Messages	123
8.3.5	Host Messages.....	135
8.3.6	Media Library Messages	138
9	Service Discovery	171
9.1	Objectives	171
9.2	Overview	171
9.3	SLP Messages	173
9.4	Scopes	174
9.5	Services Definition	175
9.5.1	Service Type.....	175
9.5.2	Service Attributes	175
9.6	User Agents (UA).....	175
9.7	Service Agents (SAs)	176
9.8	Directory Agents (DAs)	177
9.9	Service Agent Server (SA Server)	177
9.9.1	General Information.....	177
9.9.2	SA Server (SAS) Implementation.....	177
9.9.3	SA Server (SAS) Clients.....	178
9.9.4	SA Server Configuration.....	178
9.9.5	SA Server Discovery	180
9.9.6	SAS Client Registration.....	180
9.10	Configurations	180
9.10.1	Multicast Configurations	180
9.10.2	No Multicast configuration	181
9.10.3	Multicast Islands.....	182
10	Indications	185
10.1	Indications profile	185
10.1.1	Profile Support.....	185

10.1.2	Creating a client defined indication and subscription	185
10.1.3	ListenerDestination.....	185
10.2	Indication Filter Strings.....	185
10.2.1	Instance Creation	186
10.2.2	Instance Deletion.....	186
10.2.3	Modification of any value in an array property.....	186
10.2.4	Modification to either of Two Specific values in an Array Property.....	186
10.2.5	Alert	187
11	SMI-S Roles	189
11.1	Introduction	189
11.2	SMI-S Client	190
11.2.1	Overview.....	190
11.2.2	SLP Functions	190
11.2.3	Generic Operations	190
11.2.4	Security Considerations.....	190
11.2.5	Lock Management Functions	190
11.3	Dedicated SMI-S Server	190
11.3.1	Overview.....	190
11.3.2	SLP Functions	191
11.3.3	Generic Operations	191
11.3.4	Security Considerations.....	192
11.3.5	Lock Management Functions	192
11.4	General Purpose SMI-S Server	192
11.4.1	Overview.....	192
11.4.2	SLP Functions	192
11.4.3	Generic Operations	193
11.4.4	Lock Management Functions	193
11.4.5	Provider Sub-role.....	193
11.5	Directory Server	193
11.5.1	SLP Functions	193
11.5.2	Generic Operations	193
11.5.3	Security Considerations.....	193
11.5.4	Lock Management Functions	194
11.6	Combined Roles on a Single System.....	194
11.6.1	Overview.....	194
11.6.2	General Purpose SMI-S Server as a Profile Aggregator	194
12	Installation and Upgrade.....	195
12.1	Introduction	195
12.2	Role of the Administrator.....	195
12.3	Goals.....	195
12.3.1	Non-Disruptive Installation and De-installation.....	195
12.3.2	Plug-and-Play	195
12.4	Server Deployment	196
12.4.1	General.....	196
12.4.2	Controlled Environment.....	196
12.4.3	Multiple WBEM Server systems	196
12.4.4	Shared WBEM Server	197
12.4.5	Uninstallation	198
12.4.6	Update.....	198
12.4.7	Reconfiguration	198
12.5	WBEM Service Support & Related Functions	198

12.5.1	Installation	198
12.5.2	Multiple WBEM Servers on a Single Server System	199
12.5.3	Uninstallation/Upgrade	199
12.5.4	Reconfiguration	199
12.5.5	Failure.....	199
12.6	Client	199
12.6.1	Uninstallation	199
12.6.2	Reconfiguration	199
12.7	Directory Service	199
12.7.1	Installation	199
12.7.2	Uninstallation/Failure	200
12.8	Issues with Discovery Mechanisms	200
13	Security.....	201
13.1	Objectives	201
13.2	Requirements	201
13.2.1	Overview.....	201
13.2.2	General Requirements for HTTP Implementations	202
13.3	Description of SMI-S Security	202
13.3.1	Transport Security	203
13.3.2	Authentication.....	203
13.3.3	Service Discovery.....	204
Annex A	(informative) SMI-S Information Model.....	207
Annex B	(normative) Compliance with the SNIA SMI Specification.....	209

LIST OF FIGURES

Figure 1 - Experimental Maturity Level Tag	8
Figure 2 - Implemented Maturity Level Tag	8
Figure 3 - Stable Maturity Level Tag	9
Figure 4 - Deprecated Tag	9
Figure 5 - Reference Model.	42
Figure 6 - Basic Fault Detection.....	46
Figure 7 - Health Lifecycle	49
Figure 8 - Continuum	50
Figure 9 - Application Fault Region.....	51
Figure 10 - Switch Example	53
Figure 11 - Lines that Connect Classes	55
Figure 12 - iSCSI Qualified Names (iqn) Examples	69
Figure 13 - iSCSI EUI Name Example	69
Figure 14 - iSCSI 64-bit NAA Name Example.....	70
Figure 15 - iSCSI 128-bit NAA Name Example.....	70
Figure 16 - SA Server Configuration	180
Figure 17 - Multicast Configuration	181
Figure 18 - No Multicast configuration	182
Figure 19 - Multicast Islands	183
Figure 20 - SMI-S Roles	189
Figure B.1 Provider Migration	211

LIST OF TABLES

Table 1 - OperationalStatus for Disk Drive	46
Table 2 - Standard Formats for StorageVolume Names	63
Table 3 - Standard Formats for Port Names.....	64
Table 4 - Standard Formats for Storage System Names.....	66
Table 5 - Standard Operating System Names for Tape Devices.....	67
Table 6 - LogicalDisk.Name for disk partitions	68
Table 7 - GenericDiskParittion.Name for disk partitions	68
Table 8 - Standard Operating System Names for Unpartitioned Disks	68
Table 9 - Redundancy Message Arguments	71
Table 10 - Redundancy Alert Information	72
Table 11 - Environmental Message Arguments.....	72
Table 12 - Environmental Alert Information	73
Table 13 - FRU Operation Message Arguments	73
Table 14 - FRU Operation Alert Information	74
Table 15 - Password change Message Arguments	74
Table 16 - Password change Alert Information.....	74
Table 17 - User or Account Operation Message Arguments	75
Table 18 - User or Account Operation Alert Information.....	75
Table 19 - User Login Message Arguments	76
Table 20 - User Login Alert Information.....	76
Table 21 - Proxy Agent Device Communication Message Arguments	76
Table 22 - Proxy Agent Device Communication Alert Information.....	77
Table 23 - Port Status Changed Message Arguments	77
Table 24 - Port Status Changed Alert Information.....	78
Table 25 - Datacheck Error Message Arguments.....	78
Table 26 - Datacheck Error Alert Information	78
Table 27 - User Login Failure Message Arguments	79
Table 28 - User Login Failure Alert Information.....	79
Table 29 - Drive not responding Message Arguments	80
Table 30 - Drive not responding Alert Information	80
Table 31 - Fan Failure Alert Information	80
Table 32 - Power Supply Failure Alert Information	81
Table 33 - Drive Power Consumption Alert Information	81
Table 34 - Drive Voltage Alert Information.....	81
Table 35 - Predictive Failure Alert Information	82
Table 36 - Diagnostics Required Alert Information	82
Table 37 - Drive is responding Message Arguments.....	82
Table 38 - Drive is responding Alert Information	83
Table 39 - Cooling Fan Issues Cleared Alert Information.....	83
Table 40 - Power Supply Issues Cleared Message Arguments	83
Table 41 - Power Supply Issues Cleared Alert Information	84
Table 42 - Controller Failure Message Arguments	84
Table 43 - Controller Failure Alert Information.....	84
Table 44 - Controller Issues Cleared Message Arguments	84
Table 45 - Controller Issues Cleared Alert Information.....	85
Table 46 - Device Not ready Message Arguments	85
Table 47 - Error Properties for Device Not ready	86
Table 48 - Error Properties for Internal Bus Error.....	86

Table 49 - Error Properties for DMA Overflow	87
Table 50 - Error Properties for Firmware Logic Error	87
Table 51 - Front End Port Error Message Arguments	88
Table 52 - Front End Port Error Alert Information	88
Table 53 - Back End Port Error Message Arguments.....	88
Table 54 - Back End Port Error Alert Information	88
Table 55 - Remote Mirror Error Message Arguments.....	89
Table 56 - Error Properties for Remote Mirror Error	89
Table 57 - Remote Mirror Error Alert Information	89
Table 58 - Error Properties for Cache Memory Error.....	90
Table 59 - Error Properties for Unable to Access Remote Device	90
Table 60 - Error Reading Data Alert Information	91
Table 61 - Error Writing Data Alert Information	91
Table 62 - Error Validating Write (CRC) Alert Information.....	92
Table 63 - Error Properties for Copy Operation Failed	92
Table 64 - Error Properties for RAID Operation Failed.....	93
Table 65 - Error Properties for Invalid RAID Type	93
Table 66 - Error Properties for Invalid Storage Element Type.....	94
Table 67 - Error Properties for Configuration Change Failed	94
Table 68 - Error Properties for Buffer Overrun	95
Table 69 - Stolen Capacity Message Arguments	95
Table 70 - Error Properties for Stolen Capacity.....	95
Table 71 - Invalid Extent passed Message Arguments	96
Table 72 - Error Properties for Invalid Extent passed.....	96
Table 73 - Error Properties for Invalid Deletion Attempted	97
Table 74 - Error Properties for Job Failed to Start.....	97
Table 75 - Job was Halted Message Arguments	98
Table 76 - Invalid State Transition Message Arguments	98
Table 77 - Error Properties for Invalid State Transition	98
Table 78 - Invalid SAP for Method Message Arguments.....	99
Table 79 - Error Properties for Invalid SAP for Method	99
Table 80 - Resource Not Available Message Arguments	99
Table 81 - Error Properties for Resource Not Available	100
Table 82 - Resource Limit Exceeded Message Arguments.....	100
Table 83 - Error Properties for Resource Limit Exceeded	100
Table 84 - Thin Provision Capacity Warning Message Arguments	101
Table 85 - Thin Provision Capacity Warning Alert Information	101
Table 86 - Provision Capacity Critical Message Arguments.....	101
Table 87 - Provision Capacity Critical Alert Information	102
Table 88 - Thin Provision Capacity Okay Message Arguments	102
Table 89 - Thin Provision Capacity Okay Alert Information	102
Table 90 - Masking Group Membership Changed Message Arguments.....	103
Table 91 - Masking Group Membership Changed Alert Information	103
Table 92 - StorageVolume Relocation Starts Message Arguments	103
Table 93 - StorageVolume Relocation Starts Alert Information	103
Table 94 - StorageVolume Relocation Ends Message Arguments.....	104
Table 95 - StorageVolume Relocation Ends Alert Information	104
Table 96 - StoragePool Relocation Starts Message Arguments	104
Table 97 - StoragePool Relocation Starts Alert Information	104

Table 98 - StoragePool Relocation Ends Message Arguments.....	105
Table 99 - StoragePool Relocation Ends Alert Information	105
Table 100 - LogicalDisk Relocation Starts Message Arguments	105
Table 101 - LogicalDisk Relocation Starts Alert Information.....	105
Table 102 - LogicalDisk Relocation Ends Message Arguments	106
Table 103 - LogicalDisk Relocation Ends Alert Information.....	106
Table 104 - Volume or pool degraded Message Arguments	106
Table 105 - Volume or pool degraded Alert Information	106
Table 106 - Volume or pool failed Message Arguments	107
Table 107 - Volume or pool failed Alert Information.....	107
Table 108 - Volume or pool issues cleared Message Arguments.....	107
Table 109 - Volume or pool issues cleared Alert Information	108
Table 110 - The StoragePool is healthy Message Arguments.....	108
Table 111 - The StoragePool is healthy Alert Information	108
Table 112 - StoragePool is dependent on an element with problems Message Arguments	109
Table 113 - StoragePool is dependent on an element with problems Alert Information	109
Table 114 - The StoragePool is being serviced Message Arguments	110
Table 115 - The StoragePool is being serviced Alert Information.....	110
Table 116 - The OperationalStatus of the Pool is impacting an element allocated from it Message Arguments111	
Table 117 - The OperationalStatus of the Pool is impacting an element allocated from it Alert Information	111
Table 118 - The StoragePool OperationalStatus may be corrected by applying a spare Message Arguments112	
Table 119 - The StoragePool OperationalStatus may be corrected by applying a spare Alert Information.....	112
Table 120 - The StoragePool OperationalStatus may be corrected by relocating the pool Message Arguments113	
Table 121 - The StoragePool OperationalStatus may be corrected by relocating the pool Alert Information.....	113
Table 122 - Pool experiencing interference from system workload Message Arguments	114
Table 123 - Pool experiencing interference from system workload Alert Information.....	114
Table 124 - Pool performance degraded by component element Message Arguments.....	115
Table 125 - Pool performance degraded by component element Alert Information	115
Table 126 - Pool degraded due to loss of RAID protection Message Arguments.....	116
Table 127 - Pool degraded due to loss of RAID protection Alert Information	116
Table 128 - Pool degraded due to loss of port redundancy Message Arguments.....	117
Table 129 - Pool degraded due to loss of port redundancy Alert Information.....	117
Table 130 - Pool predicting failure due lack of available capacity Message Arguments	117
Table 131 - Pool predicting failure due lack of available capacity Alert Information	118
Table 132 - Zone Database Changed Message Arguments.....	118
Table 133 - Zone Database Changed Alert Information	118
Table 134 - ZoneSet Activated Message Arguments	119
Table 135 - ZoneSet Activated Alert Information	119
Table 136 - Error Properties for Session Locked.....	120
Table 137 - Error Properties for Session Aborted.....	120
Table 138 - Switch Status Changed Message Arguments	121
Table 139 - Switch Status Changed Alert Information	121
Table 140 - Fabric Merge/Segmentation Message Arguments	121
Table 141 - Fabric Merge/Segmentation Alert Information	121
Table 142 - Switch Added/Removed Message Arguments	122
Table 143 - Switch Added/Removed Alert Information	122
Table 144 - Fabric Added/Removed Message Arguments	122

Table 145 - Fabric Added/Removed Alert Information.....	122
Table 146 - Security Policy change Message Arguments	123
Table 147 - Security Policy change Alert Information	123
Table 148 - System OperationalStatus Bellwether Message Arguments	124
Table 149 - System OperationalStatus Bellwether Alert Information.....	124
Table 150 - NetworkPort OperationalStatus Bellwether Message Arguments	124
Table 151 - NetworkPort OperationalStatus Bellwether Alert Information	125
Table 152 - LogicalDisk OperationalStatus Bellwether Message Arguments.....	125
Table 153 - LogicalDisk OperationalStatus Bellwether Alert Information	125
Table 154 - CopyState is set to Broken Message Arguments	126
Table 155 - CopyState is set to Broken Alert Information.....	126
Table 156 - Not Enough Space Message Arguments.....	127
Table 157 - Not Enough Space Alert Information	127
Table 158 - The changes in RemoteReplicationCollection Message Arguments.....	127
Table 159 - The changes in RemoteReplicationCollection Alert Information.....	128
Table 160 - The changes in ProtocolEndpoint Message Arguments.....	128
Table 161 - The changes in ProtocolEndpoint Alert Information	128
Table 162 - CopyState is set to Broken Message Arguments	129
Table 163 - CopyState is set to Broken Alert Information.....	129
Table 164 - CopyState is set to Invalid Message Arguments	129
Table 165 - CopyState is set to Invalid Alert Information.....	130
Table 166 - CopyState is set to Inactive Message Arguments	130
Table 167 - CopyState is set to Inactive Alert Information.....	130
Table 168 - CopyState is set to Split Message Arguments.....	131
Table 169 - CopyState is set to Split Alert Information	131
Table 170 - CopyState alert has been cleared Message Arguments	131
Table 171 - CopyState alert has been cleared Alert Information.....	132
Table 172 - Available Space Changed Message Arguments	132
Table 173 - Available Space Changed Alert Information.....	132
Table 174 - Filesystem Inaccessible Message Arguments.....	133
Table 175 - Filesystem Inaccessible Alert Information	133
Table 176 - Filesystem is Online Message Arguments.....	133
Table 177 - Filesystem is Online Alert Information	133
Table 178 - Fileshare is degraded Message Arguments	134
Table 179 - Fileshare is degraded Alert Information.....	134
Table 180 - Fileshare in normal state Message Arguments	134
Table 181 - Fileshare in normal state Alert Information.....	135
Table 182 - Required Firmware Version Message Arguments.....	135
Table 183 - Required Firmware Version Alert Information.....	135
Table 184 - Recommended Firmware Version Message Arguments	136
Table 185 - Recommended Firmware Version Alert Information.....	136
Table 186 - Controller OK Message Arguments.....	136
Table 187 - Controller OK Alert Information.....	136
Table 188 - Controller not OK Message Arguments	137
Table 189 - Controller not OK Alert Information.....	137
Table 190 - Bus rescan complete Alert Information.....	137
Table 191 - Disk initialize Failed Message Arguments	137
Table 192 - Disk initialize Failed Alert Information.....	138
Table 193 - Read Warning Alert Information	138

Table 194 - Write Warning Alert Information.....	138
Table 195 - Hard Error Alert Information.....	139
Table 196 - Media Alert Information.....	139
Table 197 - Read Failure Alert Information.....	139
Table 198 - Write Failure Alert Information.....	140
Table 199 - Media Life Alert Information.....	140
Table 200 - Not Data Grade Alert Information.....	140
Table 201 - Write Protect Alert Information.....	141
Table 202 - No Removal Alert Information.....	141
Table 203 - Cleaning Media Alert Information.....	141
Table 204 - Unsupported Format Alert Information.....	142
Table 205 - Recoverable Snapped Tape Alert Information.....	142
Table 206 - Unrecoverable Snapped Tape Alert Information.....	142
Table 207 - Memory Chip In Cartridge Failure Alert Information.....	143
Table 208 - Forced Eject Alert Information.....	143
Table 209 - Read Only Format Alert Information.....	143
Table 210 - Directory Corrupted On Load Alert Information.....	144
Table 211 - Nearing Media Life Alert Information.....	144
Table 212 - Clean Now Alert Information.....	144
Table 213 - Clean Periodic Alert Information.....	145
Table 214 - Expired Cleaning Media Alert Information.....	145
Table 215 - Invalid Cleaning Media Alert Information.....	145
Table 216 - Retention Requested Alert Information.....	146
Table 217 - Dual-Port Interface Error Alert Information.....	146
Table 218 - Drive Maintenance Alert Information.....	146
Table 219 - Hardware A Alert Information.....	147
Table 220 - Hardware B Alert Information.....	147
Table 221 - Interface Alert Information.....	147
Table 222 - Eject Media Alert Information.....	148
Table 223 - Download Failure Alert Information.....	148
Table 224 - Loader Hardware A Alert Information.....	148
Table 225 - Loader Stray Media Alert Information.....	149
Table 226 - Loader Hardware B Alert Information.....	149
Table 227 - Loader Door Alert Information.....	149
Table 228 - Loader Hardware C Alert Information.....	150
Table 229 - Loader Magazine Alert Information.....	150
Table 230 - Loader Predictive Failure Alert Information.....	150
Table 231 - Load Statistics Alert Information.....	151
Table 232 - Media Directory Invalid at Unload Alert Information.....	151
Table 233 - Media System area Write Failure Alert Information.....	151
Table 234 - Media System Area Read Failure Alert Information.....	152
Table 235 - No Start of Data Alert Information.....	152
Table 236 - Loading Failure Alert Information.....	152
Table 237 - Library Hardware A Alert Information.....	153
Table 238 - Library Hardware B Alert Information.....	153
Table 239 - Library Hardware C Alert Information.....	153
Table 240 - Library Hardware D Alert Information.....	154
Table 241 - Library Diagnostic Required Alert Information.....	154
Table 242 - Library Interface Alert Information.....	154

Table 243 - Failure Prediction Alert Information	155
Table 244 - Library Maintenance Alert Information.....	155
Table 245 - Library Humidity Limits Alert Information.....	155
Table 246 - Library Voltage Limits Alert Information.....	156
Table 247 - Library Stray Media Alert Information.....	156
Table 248 - Library Pick Retry Alert Information.....	156
Table 249 - Library Place Retry Alert Information.....	157
Table 250 - Library Load Retry Alert Information.....	157
Table 251 - Library Door Alert Information.....	157
Table 252 - Library Mailslot Alert Information	158
Table 253 - Library Magazine Alert Information.....	158
Table 254 - Library Security Alert Information	158
Table 255 - Library Security Mode Alert Information	159
Table 256 - Library Offline Alert Information.....	159
Table 257 - Library Drive Offline Alert Information.....	159
Table 258 - Library Scan Retry Alert Information.....	160
Table 259 - Library Inventory Alert Information.....	160
Table 260 - Library Illegal Operation Alert Information	160
Table 261 - Pass Through Mechanism Failure Alert Information.....	161
Table 262 - Cartridge in Pass-through Mechanism Alert Information.....	161
Table 263 - Unreadable barcode Labels Alert Information	161
Table 264 - Throughput Threshold Warning Alert Message Arguments.....	162
Table 265 - Throughput Threshold Warning Alert Alert Information	162
Table 266 - Throughput Threshold Critical Alert Message Arguments.....	162
Table 267 - Throughput Threshold Critical Alert Alert Information.....	163
Table 268 - Physical Capacity Threshold Warning Alert Message Arguments.....	163
Table 269 - Physical Capacity Threshold Warning Alert Alert Information	163
Table 270 - Physical Capacity Threshold Critical Alert Message Arguments.....	164
Table 271 - Physical Capacity Threshold Critical Alert Alert Information.....	164
Table 272 - Logical Capacity Threshold Warning Alert Message Arguments.....	165
Table 273 - Logical Capacity Threshold Warning Alert Alert Information	165
Table 274 - Logical Capacity Threshold Critical Alert Message Arguments.....	165
Table 275 - Logical Capacity Threshold Critical Alert Alert Information.....	166
Table 276 - System Ratio Threshold Warning Alert Message Arguments.....	166
Table 277 - System Ratio Threshold Warning Alert Alert Information	166
Table 278 - System Ratio Threshold Critical Alert Message Arguments.....	167
Table 279 - System Ratio Threshold Critical Alert Alert Information.....	167
Table 280 - Deduplication Ratio Threshold Warning Alert Message Arguments.....	168
Table 281 - Deduplication Ratio Threshold Warning Alert Alert Information	168
Table 282 - Deduplication Ratio Threshold Critical Alert Message Arguments.....	168
Table 283 - Deduplication Ratio Threshold Critical Alert Alert Information.....	169
Table 284 - Replication Traffic Threshold Warning Alert Message Arguments	169
Table 285 - Replication Traffic Threshold Warning Alert Alert Information.....	169
Table 286 - Replication Traffic Threshold Critical Alert Message Arguments.....	170
Table 287 - Replication Traffic Threshold Critical Alert Alert Information	170
Table 288 - Message Types	174
Table 289 - Required Configuration Properties for SA as DA.....	178
Table 290 - Required Configuration Properties for SA	179
Table 291 - Create an IndicationFilter and subscribe to it	185

FOREWORD

Storage Management Technical Specification, Part 2 Common Architecture, 1.8.0 Rev 3 defines the core architecture of SMI-S. This includes the protocols (WBEM, SLP,...); the model is defined in the other specification parts.

Parts of this Standard

This standard is subdivided in the following parts:

- *Storage Management Technical Specification, Part 1 Overview, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 2 Common Architecture, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 4 Block Devices, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 5 Filesystems, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 6 Fabric, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 7 Host Elements, 1.8.0 Rev 3*
- *Storage Management Technical Specification, Part 8 Media Libraries, 1.8.0 Rev 3*

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 2 Common Architecture, 1.8.0 Rev 3 defines the core architecture and protocols in SMI-S. The components of SMI-S architecture include:

- Transport - communicating management information between constituents of the management system
- Health and fault management - detecting failures through monitoring the state of storage components
- General information about the object model
- Names - how SMI-S uses names to allow applications to correlate across SMI-S and to other standards
- Standard messages - how exceptions are presented to client applications
- Service discovery - techniques clients use to discover SMI-S services
- Installation and upgrade - recommendations for implementations
- Compliance - requirement for compliance to the standard

2 Normative references

2.1 General

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.

2.2 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)

SNIA TLS Specification for Storage Systems

http://www.snia.org/sites/default/files/TLSspec-v1.01_Technical_Position.pdf

2.3 DMTF references (Final)

DMTF Final documents are accepted as standards. For DMTF Draft or Preliminary documents, see 2.5.

DMTF DSP0004, CIM Infrastructure Specification 3.0.1

http://www.dmtf.org/sites/default/files/standards/documents/DSP0004_3.0.1.pdf

DMTF DSP0200, CIM Operations over HTTP 1.4

http://www.dmtf.org/sites/default/files/standards/documents/DSP0200_1.4.pdf

DMTF DSP0201 Representation of CIM in XML 2.4

http://www.dmtf.org/sites/default/files/standards/documents/DSP0201_2.4.pdf

DMTF DSP0202 CIM Query Language Specification 1.0

http://www.dmtf.org/standards/published_documents/DSP0202_1.0.0.pdf

DMTF DSP0205 WBEM Discovery Using the Service Location Protocol 1.0.1

http://dmtf.org/sites/default/files/standards/documents/DSP0205_1.0.1.pdf

DMTF DSP0206 WBEM SLP Template 1.0

<http://www.dmtf.org/sites/default/files/standards/documents/wbem.1.0.en>

DMTF DSP0207, 1.0.1 WBEM URI Mapping Specification

http://www.dmtf.org/sites/default/files/standards/documents/DSP0207_1.0.1.pdf

DMTF DSP0210 CIM-RS Protocol 1.0

http://www.dmtf.org/sites/default/files/standards/documents/DSP0210_1.0.pdf

DMTF DSP0211 CIM-RS Payload Representation in JSON 1.0

http://www.dmtf.org/sites/default/files/standards/documents/DSP0211_1.0.pdf

DMTF DSP0221 Managed Object Format (MOF) 3.0.1

http://www.dmtf.org/sites/default/files/standards/documents/DSP0221_3.0.1.pdf

DMTF DSP0223 Generic Operations 2.0.0

http://www.dmtf.org/sites/default/files/standards/documents/DSP0223_2.0.0.pdf

DMTF DSP0226, WS-Management Protocol Specification 1.1.1

http://www.dmtf.org/sites/default/files/standards/documents/DSP0226_1.1.1.pdf

DMTF DSP0227, WS-Management CIM Binding Specification 1.2

http://www.dmtf.org/sites/default/files/standards/documents/DSP0227_1.2.pdf

DMTF DSP0228, Message Registry Schema, 1.1.0
http://schemas.dmtf.org/wbem/messageregistry/1/dsp0228_1.1.0.xsd

DMTF DSP2011, Standard Messages Whitepaper 1.0
<http://www.dmtf.org/sites/default/files/standards/documents/DSP2011.pdf>

DMTF DSP0230, WS-CIM Mapping Specification 1.1.0
http://www.dmtf.org/sites/default/files/standards/documents/DSP0230_1.1.0.pdf

DMTF DSP8000 1.2.0 Message Registry Print XSLT Stylesheet
http://schemas.dmtf.org/wbem/messageregistry/1/dsp8000_1.2.0.xsl

DMTF DSP8016 2.0.0 WBEM Operations Message Registry
http://schemas.dmtf.org/wbem/messageregistry/1/dsp8016_2.0.0.xml

2.4 IETF references

For IETF Informational documents and proposed standards, see 2.5.

IETF RFC 2045, Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies
<http://www.ietf.org/rfc/rfc2045.txt>

IETF RFC 2246, The TLS Protocol Version 1.0
<http://www.ietf.org/rfc/rfc2246.txt>

IETF RFC 4291, IP Version 6 Addressing Architecture

IETF RFC 2396, Uniform Resource Identifiers (URI)
<http://www.ietf.org/rfc/rfc2396.txt>

IETF RFC 2608, Service Location Protocol, Version 2
<http://www.ietf.org/rfc/rfc2608.txt>

IETF RFC 2609, Service Templates and Service: Schemes
<http://www.ietf.org/rfc/rfc2609.txt>

IETF RFC 2610, DHCP Options for Service Location Protocol
<http://www.ietf.org/rfc/rfc2610.txt>

IETF RFC 2616, Hypertext Transfer Protocol -- HTTP/1.1
<http://www.ietf.org/rfc/rfc2616.txt>

IETF RFC 2617, HTTP Authentication: Basic and Digest Access Authentication
<http://www.ietf.org/rfc/rfc2617.txt>

IETF RFC 2445, Internet Calendaring and Scheduling Core Object Specification (iCalendar)
<http://www.ietf.org/rfc/rfc2445.txt>

IETF RFC 3280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile
<http://www.ietf.org/rfc/rfc3280.txt>

IETF RFC 3723, Securing Block Storage Protocols over IP
<http://www.ietf.org/rfc/rfc3723.txt>

IETF RFC 3986, Definitions of Managed Objects for the DS3/E3 Interface Type
<http://www.ietf.org/rfc/rfc3986.txt>

IETF RFC 4291, IP Version 6 Addressing Architecture
<http://www.ietf.org/rfc/rfc4291.txt>

IETF RFC 4514, Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names
<http://www.ietf.org/rfc/rfc4514.txt>

IETF RFC 5246, The Transport Layer Security (TLS) Protocol Version 1.2
<http://tools.ietf.org/rfc/rfc5246.txt>

2.5 References under development

The following documents (and their web addresses) are subject to change.

DMTF DSP8055 1.0.0d Diagnostic Message Registry
http://www.dmtf.org/sites/default/files/standards/documents/DSP8055_1.0.0d.xml

2.6 Other references

IETF RFC 1945 Hypertext Transfer Protocol -- HTTP/1.0
<http://www.ietf.org/rfc/rfc1945.txt>

IETF RFC 2614 An API for Service Location
<http://www.ietf.org/rfc/rfc2614.txt>

UML (Universal Modeling Language) Specifications
http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML

ITU-T Recommendation X.509 (1997 E): Information Technology - Open Systems Interconnection - The Directory: Authentication Framework

PKCS #12, Personal Information Exchange Syntax
<http://www.rsasecurity.com/rsalabs/node.asp?id=2138>

3 Terms, definitions, symbols, abbreviations, and conventions

3.1 Terms and definitions

3.1.1

access control

means to ensure authorized access and to prevent unauthorized access to resources relevant to information security based on the business and security requirements

3.1.2

account

an established relationship between a user and a computer, network or information service

3.1.3

accountability

Information Security property that establishes responsibility for the effects of action taken by individuals, organizations or communities with an explanation as to how and why the action took place.

[SOURCE: ISO/IEC 27000]

3.1.4

administrator

a person charged with the installation, configuration, and management of a computer system, network, storage subsystem, database, or application

3.1.5

agent

an Object Manager that includes the provider service for a limited set of resources

3.1.6

aggregation

a strong form of an association

3.1.7

audit Log

logs collecting the evidence of selected user activities, exceptions, and information security events

3.1.8

authentication

the act of verifying the identity claimed by a party to a communication

3.1.9

authentication mechanism

process for determining and validating a user (or device) identity

3.1.10

authorization

the process of granting a right or permission to access a system resource

3.1.11

bidirectional authentication

See "mutual authentication"

3.1.12

CIM Server

See "WBEM Server".

client

a process that issues requests for service

3.1.13

Common Information Model

an object-oriented description of the entities and relationships in a business' management environment maintained by the Distributed Management Task Force

3.1.14

dedicated SMI-S Server

a WBEM Server that is dedicated to supporting a single device or subsystem

3.1.15

digested password

the hashed form of a cleartext password

3.1.16

discovery

a process which provides information about what physical and logical storage entities have been found within the management domain

3.1.17

Distributed Management Task Force (DMTF)

an industry organization that develops management standards for computer system and enterprise environments

3.1.18

dynamic host control protocol (DHCP)

an Internet protocol that allows nodes to dynamically acquire ("lease") network addresses for periods of time rather than having to pre-configure them

3.1.19

embedded SMI-S Server

a WBEM Server that is embedded in the device or subsystem for which it provides management

3.1.20

enclosure

a box or cabinet

3.1.21

entity authentication

corroboration that an entity is the one claimed.
[SOURCE: ISO/IEC 9798]

3.1.22

enumerate

an operation used to enumerate subclasses, subclass names, instances and instance names

3.1.23

event

an occurrence of a phenomenon of interest.

3.1.24

eXtensible Markup Language

a universal format for structured documents and data on the World Wide Web

3.1.25

extent

a set of consecutively addressed disk blocks.

3.1.26

external authentication

authentication which relies on an authentication service separate from (or external to) an entity

3.1.27

extrinsic method

A method defined as part of CIM Schema

3.1.28

fabric

Any interconnect between two or more Fibre Channel N_Ports, including point-to-point, loop, and Switched Fabric.

3.1.29

FICON™¹

Fibre Channel storage protocol used in IBM mainframe computers and peripheral devices such as ECKD storage arrays and tape drives

3.1.30

general purpose SMI-S Server

an SMI-S Server that is not dedicated to supporting a single device or subsystem, and may support multiple devices or subsystems.

3.1.31

grammar

a formal definition of the syntactic structure of a language (see "syntax"), normally given in terms of production rules that specify the order of constituents and their sub-constituents in a sentence (a well-formed string in the language)

3.1.32

host bus adapter (HBA)

card that contains ports for host systems

3.1.33

Hypertext Transfer Protocol (HTTP)

request-reply protocol used for internet communications

3.1.34

identity

representation of an actual user (or application or service or device)

3.1.35

interconnect element

non-terminal network elements (Switches, hubs, routers, directors).

3.1.36

interface definition language (IDL)

high-level declarative language that provides the syntax for interface declarations

3.1.37

intrinsic method

operations made against a WBEM server and a CIM namespace independent of the implementation of the schema defined in the server

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.

3.1.38

logical unit number

a SCSI logical unit or logical unit number.

3.1.39

mutual authentication

authentication that provides both parties (users or entities) with assurance of each other's identity.

3.1.40

Network Address Authority (NAA)

a four bit identifier to denote a network address authority (i.e., an organization such as CCITT or IEEE that administers network addresses)

3.1.41

non-repudiation

the ability to prove an action or event has taken place, so that this event or action cannot be repudiated later

[SOURCE: ISO/IEC 27000]

3.1.42

out-of-band

transmission of management information for storage components outside of the data path, typically over Ethernet.

Also: use of mechanisms other than the ones required on a communications channel to transmit information.

[SOURCE: ISO/IEC 24767]

3.1.43

partition

collection of contiguous block on a disk or virtual disk

3.1.44

password

a secret sequence of characters or a word that a user submits to a system for purposes of authentication, validation, or verification.

3.1.45

path

combination of initiator and target ports and logical unit

3.1.46

privacy

the right of an entity (normally an individual or an organization), acting on its own behalf, to determine the degree to which the confidentiality of their private information is maintained.

3.1.47

privileged user

a user who, by virtue of function, and/or seniority, has been allocated powers within a system, which are significantly greater than those available to the majority of users

3.1.48

protocol

a set of rules that define and constrain data, operations, or both

3.1.49

proxy SMI-S Server

an SMI-S Server that does not run on the device or subsystem which it supports

3.1.50

public key infrastructure (PKI)

a framework established to issue, maintain, and revoke public key certificates accommodating a variety of security technologies.

3.1.51

SAN

a group of fabrics that have common leaf elements.

3.1.52

SCSI Parallel Interface (SPI)

The family of SCSI standards that define the characteristics of the parallel version of the SCSI interface.

3.1.53

Secure Sockets Layer (SSL)

A suite of cryptographic algorithms, protocols and procedures used to provide security for communications used to access the world wide web.

Note 1 to entry: More recent versions of SSL are known as TLS (Transport Level Security) and are standardized by the Internet Engineering Task Force (IETF).

3.1.54

Service Access Point

the network address of a process offering a service.

3.1.55

shared secret

a pre-shared key that has been allocated to communicating parties prior to the communication process starting.

3.1.56

Simple Network Management Protocol (SNMP)

an IETF protocol for monitoring and managing systems and devices in a network

3.1.57

SMI-S server

a WBEM Server that supports SMI-S (*Storage Management Initiative Specification*) profiles for management of a device or subsystem

3.1.58

SNMP trap

a type of SNMP message used to signal that an event has occurred

3.1.59

soft zone

a zone consisting of zone Members that are made visible to each other through Client Service requests

3.1.60

Storage Management Initiative Specification (SMI-S)

an interface between WBEM-capable clients and servers for the secure, extensible, and interoperable management of networked storage (this standard)

3.1.61

Storage Networking Industry Association (SNIA)

an association of producers and consumers of storage networking products whose goal is to further storage networking technology and applications

3.1.62

storage resource management (SRM)

management of physical and logical storage resources, including storage elements, storage devices, appliances, virtual devices, disk volume and file resources

3.1.63

switch

fibre channel interconnect element that supports a mesh topology

3.1.64

switched fabric

a fabric comprised of one or more Switches

3.1.65

syntax

the structure of strings in some language. A language's syntax is described by a grammar

3.1.66

threat

a potential source of an incident that may result in adverse changes to an asset, a group of assets or an organization

[SOURCE: ISO/IEC 27000]

3.1.67

unidirectional authentication

authentication that provides one party (user or entity) with assurance of the other's identity

3.1.68

User Datagram Protocol

an Internet protocol that provides connectionless datagram delivery service to applications

3.1.69

vulnerability

weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat

[SOURCE: ISO/IEC 27000]

3.1.70

Web Based Enterprise Management (WBEM)

a set of management and Internet standard technologies from DMTF developed to unify the management of distributed computing environments

3.1.71

WBEM Server

a server that provides support for CIM requests and provides CIM responses using web protocols

3.1.72

web service

a software system designed to support interoperable machine-to-machine interaction over a network

3.1.73

zone

a group of ports and switches that allow access. Defined by a zone definition

3.1.74

zone set

one or more zones that may be activated or deactivated as a group

3.2 Acronyms and abbreviations

AAA	Authentication, Authorization, and Accounting
ACL	Access Control List
API	application programming interface
CA	Certificate Authority
CIM	Common Information Model
CRL	Certificate Revocation List
DHCP	dynamic host control protocol
FC	Fibre Channel
HBA	host bus adapter
HMAC	keyed-Hash Message Authentication Code
HTTPS	Hypertext Transfer Protocol Secure
IDL	interface definition language
IETF	Internet Engineering Task Force
IMA	iSCSI Management API
IP	Internet Protocol
IPsec	Internet Protocol Security
iSCSI	Internet SCSI
LDAP	Lightweight Directory Access Protocol
MAC	Message Authentication Code
OS	operating system
PKI	Public Key Infrastructure
RADIUS	Remote Authentication Dial In User Service
RBAC	Role-base Access Control
RFC	Request for Comments
SAM-3	SCSI Architecture Model
SAN	storage area network
SB	Single Byte (command set)
SCSI	Small Computer System Interface
SES	SCSI Enclosure Services
SLP	Service Location Protocol
SMI-S	Storage Management Initiative - Specification
SNIA	Storage Networking Industry Association
SPC-3	SCSI Primary Commands-3
SSL	Secure Socket Layer
SSO	Single Sign-on
SSP	Storage Service Provider
TC	Technical Committee
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
URL	Uniform Resource Locator
WBEM	Web-Based Enterprise Management
XML	Extensible Markup Language

3.3 Keywords

expected

a keyword used to describe the behavior of the hardware or software in the design models presumed by this standard

Other hardware and software design models may also be implemented.

3.3.1

invalid

a keyword used to describe an illegal or unsupported bit, byte, word, field or code value

Note 1 to entry: Receipt of an invalid bit, byte, word, field or code value shall be reported as an error.

3.3.2

mandatory

a keyword indicating an item that is required to be implemented as defined in this standard to claim compliance with this standard.

3.3.3

may

a keyword that indicates flexibility of choice with no implied preference

3.3.4

may not

keywords that indicates flexibility of choice with no implied preference

3.3.5

obsolete

a keyword indicating that an item was defined in prior standards but has been removed from this standard

3.3.6

opaque

a keyword indicating that value has no semantics or internal structure

3.3.7

optional

a keyword that describes features that are not required to be implemented by this standard

Note 1 to entry: However, if any optional feature defined by this standard is implemented, it shall be implemented as defined in this standard.

3.3.8

reserved

a keyword referring to bits, bytes, words, fields and code values that are set aside for future standardization

Note 1 to entry: Their use and interpretation may be specified by future extensions to this or other standards.

Note 2 to entry: A reserved bit, byte, word or field shall be set to zero, or in accordance with a future extension to this standard. Recipients are not required to check reserved bits, bytes, words or fields for zero values.

Note 3 to entry: Receipt of reserved code values in defined fields shall be reported as an error.

3.3.9

shall

a keyword indicating a mandatory requirement

Designers are required to implement all such requirements to ensure interoperability with other products that conform to this standard.

3.3.10

should

a keyword indicating flexibility of choice with a preferred alternative; equivalent to the phrase "it is recommended"

3.4 Conventions

Certain words and terms used in this American National Standard have a specific meaning beyond the normal English meaning. These words and terms are defined either in 3 Terms, definitions, symbols, abbreviations, and conventions or in the text where they first appear.

Numbers that are not immediately followed by lower-case b or h are decimal values.

Numbers immediately followed by lower-case b (xxb) are binary values.

Numbers immediately followed by lower-case h (xxh) are hexadecimal values.

Hexadecimal digits that are alphabetic characters are upper case (i.e., ABCDEF, not abcdef).

Hexadecimal numbers may be separated into groups of four digits by spaces. If the number is not a multiple of four digits, the first group may have fewer than four digits (e.g., AB CDEF 1234 5678h)

Decimal fractions are initiated with a comma (e.g., two and one half is represented as 2,5).

Decimal numbers having a value exceeding 999 are separated with a space(s) (e.g., 24 255).

See also “ Typographical Conventions “(in front matter) for typographical conventions.

4 Transport and Reference Model

4.1 Introduction

4.1.1 Overview

The interoperable management of storage devices and network elements in a distributed storage network requires a common transport for communicating management information between constituents of the management system. This section of the specification details the design of this transport, as well as the roles and responsibilities of constituents that use the common transport (i.e., a reference model).

4.1.2 Language Requirements

To express management information across the interface, a language is needed that:

- Can contain platform independent data structures,
- Is self describing and easy to debug,
- Can be extended easily for future needs.

4.1.3 Communications Requirements

Communications protocols to carry the XML based management information are needed that:

- Can take advantage of the existing ubiquitous IP protocol infrastructures,
- Can be made to traverse inter- and intra-organizational firewalls,
- Can easily be embedded in low cost devices.

The Hyper Text Transport Protocol (HTTP) was chosen for the messaging protocol and TCP was chosen for the base transfer protocol to carry the XML management information for this interface as they meet the requirements in 4.1.3.

4.1.4 XML Message Syntax and Semantics

In order to be successful, the expression of XML management information (messages) across this interface needs to follow consistent rules for semantics and syntax. These rules are detailed in this specification. They are of sufficient quality, extensibility, and completeness to allow their wide adoption by storage vendors and management software vendors in the industry. In addition, to facilitate rapid adoption, existing software that can parse, marshal, un-marshal, and interpret these XML messages should be widely available in the market such that vendor implementations of the interface are accelerated. The message syntax and semantics selected should:

- Be available on multiple platforms,
- Have software implementations that are Open source (i.e., collaborative code base),
- Have software implementations available in Java and C++,
- Leverage industry standards where applicable,
- Conform with W3C standards for XML use.
- Be object model independent (i.e., be able to express any object model).

Virtually the only existing industry standard in this area is the WBEM standards as developed and maintained by the DMTF.

4.2 Transport Stack

It is the primary objective of this interface to drive seamless interoperability across vendors as communications technology and the object model underlying this interface evolves. Accordingly, the transport stack has been layered such that (if required) other protocols can be added as technology evolves. For example, should SOAP or IIOB become prominent, the content in the stack could be expanded with minimal changes to existing product implementations in the market. This specification relies on the DMTF WBEM Protocol Specifications.

To be compliant with this specification, CIM-XML shall be supported.

Optionally, other WBEM protocols, such as WS-Management or CIM-RS, may additionally be supported.

It should be noted that this specification places no restriction on the physical network selected to carry this transport stack. For example, a vendor can choose to use in-band communication over Fibre-channel as the backbone for this interface. Another vendor could exclusively (and wisely) choose out-of-band communication over Ethernet to implement this management interface. Additionally, select vendors could choose a mix of in-band and out-of-band physical network to carry this transport stack.

4.3 Reference Model

4.3.1 Overview

As shown in Figure 7, the Reference Model shows all possible constituents of the management environment in the presence of the transport stack for this interface.

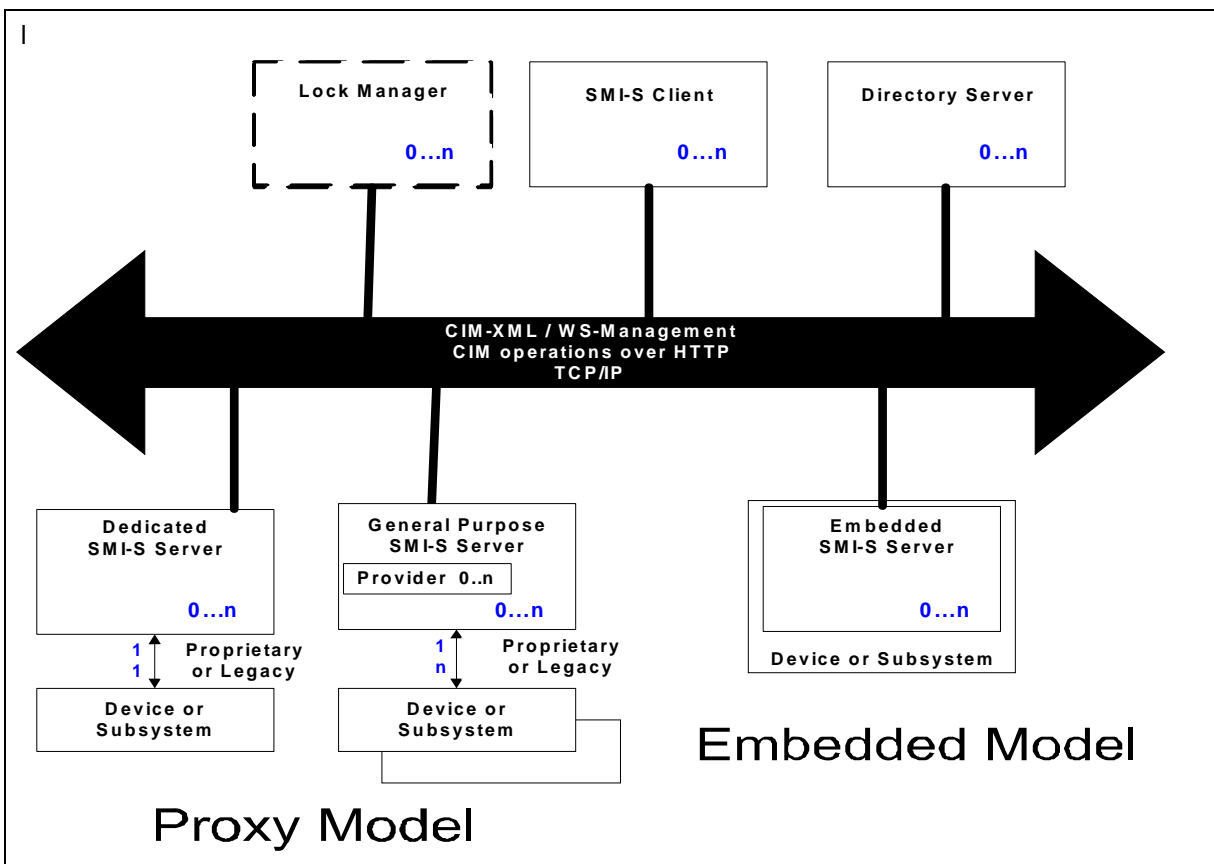


Figure 7 -Reference Model.

Figure 7 illustrates that the transport for this interface uses a WBEM Protocol to execute intrinsic and extrinsic methods against the schema for this interface.

NOTE It is envisioned that a more complete version of this reference model would include a Lock Manager. However, in this version of SMI-S, a Lock Manager is not specified. As a result, it is shown as a dotted box to illustrate where the role would fit.

4.3.2 Roles for Interface Constituents

4.3.2.1 Client

A Client is the consumer of the management information in the environment. It provides an API (language binding in Java or C++ for example) for overlying management applications (like backup engines, graphical presentation frameworks, and volume managers) to use.

4.3.2.2 SMI-S Server

An SMI-S Server is a WBEM Server. Often, an SMI-S Server controls only one device or subsystem, and is incapable of providing support for complex intrinsic methods like schema traversal. An SMI-S Server can be embedded in a device (like a Fibre Channel Switch) or provide a proxy on a host that communicates to a device over a legacy or proprietary interconnect (like a SCSI based array controller).

Embedding an SMI-S Server directly in a device or subsystem reduces the management overhead seen by a customer and eliminates the requirement for a stand-alone host (running the proxy agent) to support the device.

Embedded SMI-S Servers are the desired implementation for “plug and play” support in an SMI-S managed environment. However, proxy SMI-S Servers are a practical concession to the legacy devices that are already deployed in storage networked environments. In either case, the minimum CIM support for SMI-S Servers applies to either SMI-S Server deployments.

4.3.2.3 General Purpose SMI-S Server

A General Purpose SMI-S Server serves management information from one or more devices or underlying subsystems through providers. As such a General Purpose SMI-S Server is an aggregator that enables proxy access to devices/subsystems and can perform more complex operations like schema traversals. A General Purpose SMI-S Server typically includes a standard provider interface to which device vendors adapt legacy or proprietary product implementations.

4.3.2.4 Provider

A provider expresses management information for a given resource such as a storage device or subsystem exclusively to a WBEM Server. The resource may be local to the host that runs the Object Manager or may be remotely accessed through a distributed systems interconnect.

4.3.2.5 Lock Manager

This version of the specification does not support a lock manager.

4.3.2.6 Directory Server (SLP Directory Agent)

A directory server provides a common service for use by clients for locating services in the management environment.

4.3.3 Cascaded Agents

This specification discusses constituents in the SMI-S environment in the context of Clients and Servers. This version of the specification also allows constituents in a SMI-S management environment to function as both client and server.

Directory Server (SLP Directory Agent)

5 Health and Fault Management

5.1 Objectives

Health and Fault Management is the activity of anticipating or detecting failures through monitoring the state of the storage network and its components and intervening before services can be interrupted. A service in this case is the realization of storage through several interconnected devices connected, configured for a dedicated purpose. The purpose is the delivery of software application functionality in support of some business function.

5.2 Overview

- Express states and statuses with standard meanings.
- Define the use of comprehensive error reporting in determining the type, category, and source of failures.
- Define the quality associated with errors rather than qualities.
- Define explicit failure scopes rather than requiring HFM enabled application to construct them.

5.3 Terms

5.3.1 error

An unexpected condition, result, signal or datum. An error is usually caused by an underlying problem in the system such as a hardware fault or software defect. Errors can be classified as correctable (recoverable) or uncorrectable, detectable or undetectable.

5.3.2 fault

A problem that occurs when something is broken and therefore not functioning in the manner it was intended to function. A fault may cause an error to occur.

5.3.3 fault region

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 affect the corrective action being attempted.

5.3.4 Health and Fault Management (HFM)

Health and Fault Management is the activity of anticipating or detecting debilitating failures through monitoring the state of the storage network and its components and intervening in before services can be interrupted. A service in this case is the realization of storage utilization through several interconnected devices connected, configured for a dedicated purpose. The purpose is the delivery of software application functionality in support of some business function.

5.3.5 operational status

These values indicate the current status(es) of the element. Various operational statuses are defined (e.g., OK, starting, stopping, stopped, In Service, No Contact).

5.3.6 health state

These values indicate the current health of the element. This attribute expresses the health of this element but not necessarily that of its subcomponents.

5.4 Description of Health and Fault Management

The goal of effective administration requires devices and applications that comprise storage services to report their status and the nature of their errors in standard terms. These terms need to be understandable by a client without device-specific knowledge.

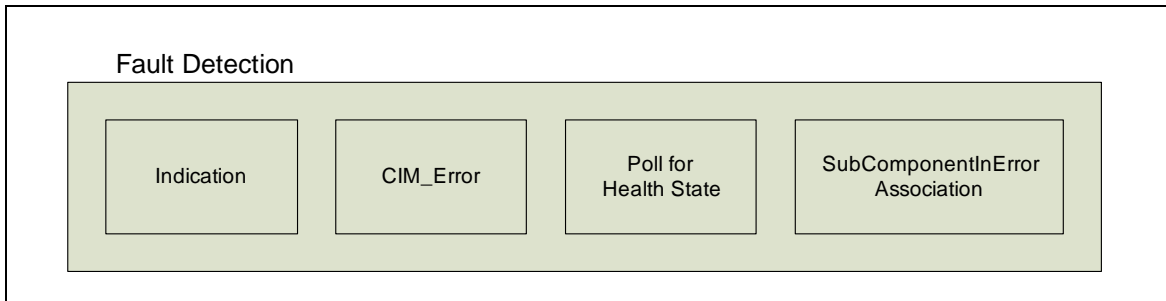


Figure 8 - Basic Fault Detection

There are four basic ways for a SMI-S client to detect an error or fault condition. Figure 8 lists the four basic methods for fault detection. These are:

- Health state and Operational status - Polling.
- Error - Standard errors returned from CIM operations.
- Indications - Subscribe for and receive asynchronous Indications.
- Fault Regions (experimental) - Walk the CIM model looking for RelatedElementCausingError associations.

5.4.1 Operational Status and Health State (Polling)

Operational Status and Health State are the two properties that will be used to monitor health. These two properties could convey very different statuses and may at times be related or independent of each other. For example, you may have a disk drive with the Operational Status of “Stopped” and the HealthState of 30 (Non-recoverable error) or 5 (OK). Now the reason the disk drive is stopped could vary from the fact that it had a head crash (HealthState = 30) to the situation where it was stopped for the routine maintenance (HealthState = 5).

Table 1 is an example of how HealthState can disambiguate health for a disk drive, various values for OperationalStatus and HealthState:

Table 1 shows, for a disk drive, various possible values for OperationalStatus and HealthState. Note that there are many cases not shown.:

Table 1 - OperationalStatus for Disk Drive

Operational Status	Description	HealthState	Description	Comment
2	OK	5	OK	Everything is fine
2	OK	10	Degraded/Warning	Some soft errors
3 or 5	Degraded or Predicted Failure	15	Minor Failure	Many soft errors
3 or 5	Degraded or Predicted Failure	20	Major Failure	Some hard errors
3	Degraded	10	Good	A subcomponent has failed (no data loss)

Table 1 - OperationalStatus for Disk Drive (Continued)

Operational Status	Description	HealthState	Description	Comment
10	Stopped	5	OK	Drive spun down normally
10	Stopped	30	Non-recoverable Error	Head crash
8	Starting	10	Degraded/Warning	Will update HealthState once fully started
4	Stressed	5	OK	Too many I/O in progress, but the drive is fine.
15	Dormant	5	OK	The drive is not needed currently

The property OperationalStatus is multi-valued and more dynamic. It tends to emphasize the current status and potentially the immediate status leading to the current status; whereas, the property HealthState is less dynamic and tends to imply the health over a longer period of time. Again, in the disk drive example, the disk drive's operational status may change many times in a given time period. However, in the same time period, the health of the same drive may not change at all.

5.4.2 Standard Errors and Events

Standardization of error and events are required so that the meaning is unambiguous and is given to comparisons.

Error and Alert indications

HFM clients shall not be required to be embodied with specific knowledge of the devices and applications in order to derive the quality of the error from the datum. The device and application shall express the quality of the error rather than the quantity interpreted with *a priori* knowledge to determine that error condition is present. For example, a device needs to express that it is too hot rather than requiring the HFM enabled application to determine this from the temperature datum and device specific knowledge of acceptable operating conditions.

Standard errors are defined for each Profile. The definitions will be contained in the profiles. Standard errors are not the only error codes that can be returned, but are the only codes that a generic client will understand.

5.4.3 Indications

Indications are asynchronous messages from WBEM Servers to clients. A client must register for them. Each SMI-S profile contains lists of indication filters that clients use to indicate the information it is interested in. The message itself is defined in the SMI-S Indications Profile (in *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3*).

EXPERIMENTAL

5.4.4 Event Correlation and Fault Containment

Automation will require that an error arising through control and configuration activities, as a side effect of them, or by failures caused by defects can be directly correlatable. Error categories like network cabling failures or network transmission errors will help organize the types of error that can be produced. Standard errors, like impending disk media failure, will be required as well.

Once the errors have been collected and correlated, the HFM enabled application can produce an impact list sorted by likelihood. Some of the error correlation can be determined by the common affect through

the manifestation of the RelatedElementCausingError association to be described later. The alerts themselves can report its correlation with other alerts.

Potential faults can then be derived from errors for each component. Deriving such a list may require a dialog between the HFM enabled application and the device or application in question such that the HFM enabled application is assisted in the production of the list.

If permitted, then control and configuration operations may be executed to contain the fault. The pallet of these operations will be those operations already available through SMI-S. However, special operations may arise from the HFM design work as well. Fault containment will include the reconfiguration of the storage service with alternative components, leaving failing components or interconnections isolated.

Much like a physician, the HFM enabled application is notified or consulted when symptoms appear. The HFM enabled application then develops a prognosis based on the manifestation of the ailment. At times, the HFM enabled application will perform diagnostic procedures. The end result of the process is to produce a list of possible causes, ranked by probability, and associated recommended procedures.

Also like a doctor, the HFM enabled application will settle for enabling the patients to heal themselves. That is the HFM enabled applications cannot be expected to heal the device in all cases. A significant portion of all possible corrective actions will require the intervention of people or device unique knowledge.

The simplified state diagram shown in Figure 9 follows the fault mitigation life cycle for HFM.

The device or application manifests an event, either by a state change, error returned from a WBEM operation, or an alert indication.

The event is recognized by the HFM enabled application and accessed by the HFM enabled application. It may be that the event indication does the represent the existence of an error. An error condition may be heralded by a single or multiple events occurring in some order. The process of examining and characterizing event as errors is called error handling.

Once it is determined that an error condition is present, then possible causes are sought and ranked by likelihood. The causes themselves describe a potential problem or fault with the component in question. Alternatively, the device or application may report the fault directly, through an alert indication, optionally with recommended actions.

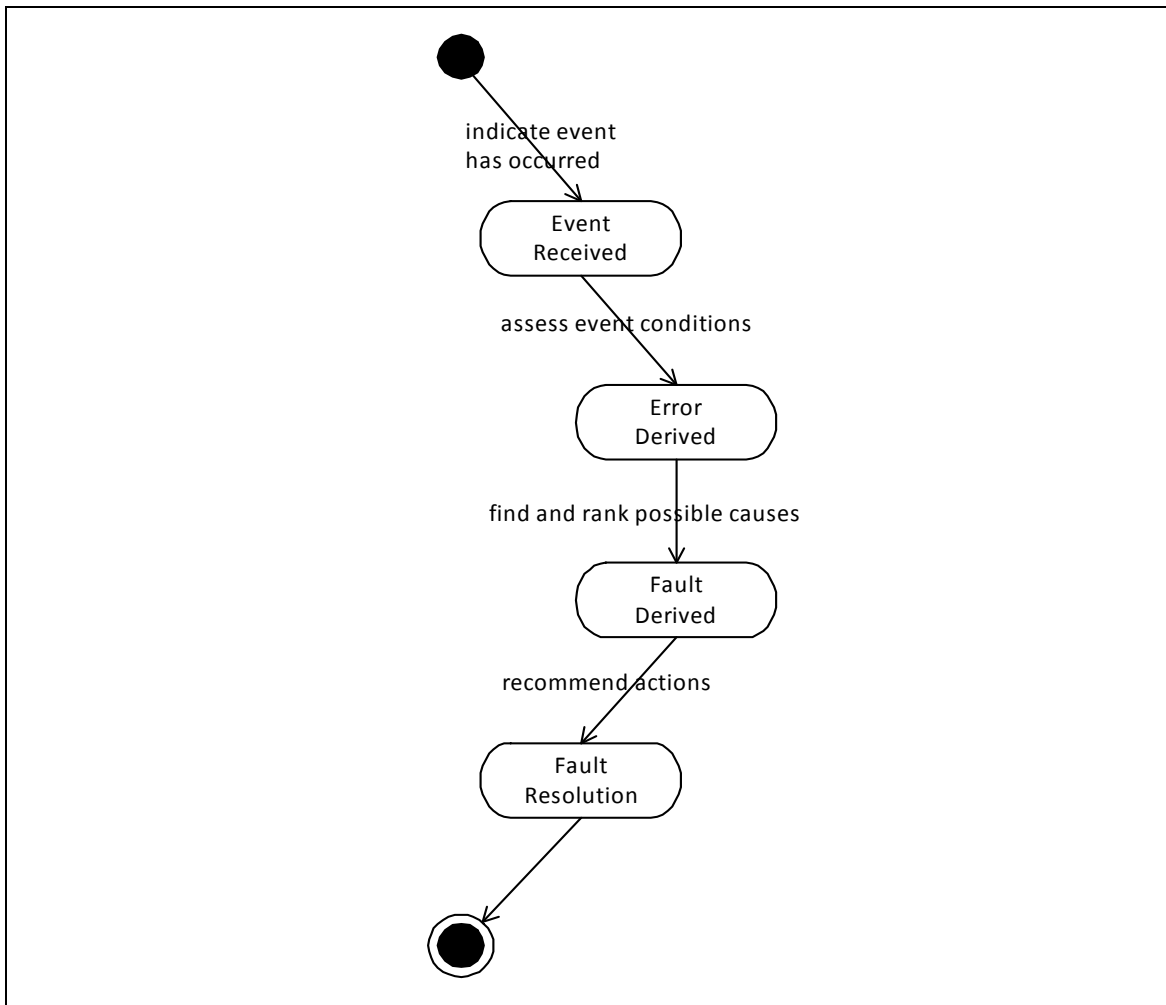


Figure 9 - Health Lifecycle

Fault resolution may not require the intervention of an operator or field technician. It is these faults that can be handled entirely by the HFM enabled application. Otherwise, the HFM enabled application cannot actively participate in whole fault resolution life cycle. In this case, the HFM enabled application would wait for the end state of fault resolution to come to being before ending its fault mitigation exercise.

Faults are contained and components repaired or replaced. The instructions to the HFM enabled application for what can be done to repair the fault are the recommended actions. Fault Containment includes fencing off the faulty component and maintaining the service. To be minimally effective, the HFM enabled application contains the fault. The repair may or may not be done with human intervention.

The devices and application that comprise a storage system have themselves some level of self diagnostics and report functionality.

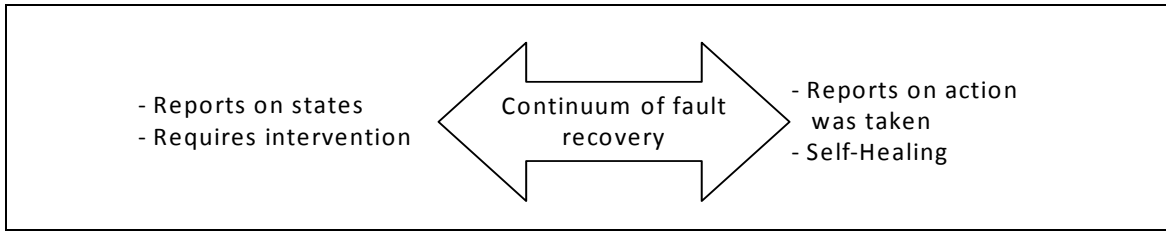


Figure 10 - Continuum

There is a range of ability of devices and applications to recover from failures and to report on the error recovery actions taken. See Figure 10. The variance of capabilities for device and applications can be plotted on a continuum. At one end of continuum, the device or application recognizes a fault condition and takes action, reporting on the action taken and any further action required to service it. At the other end of the continuum, the device can only report on that states and requires intervention both in the detection of fault conditions and taking corrective action.

There are limits to what an HFM enabled application can do. Obviously, if the device or application cannot report states, errors and alerts in a standard way or cannot report this data at all, then there is little an external implementation can do.

However, few, if any, of these devices and applications can monitor and correct the service as a whole. It is for this reason, the HFM implementation is needed to augment the effectiveness of the administrator.

5.4.5 Fault Regions

A scope can be applied to the effect of errors and the associated fault. A fault may affect a component, a device or application, storage service, or all the above. This scope defines the area of influence for fault containment. For example, the device itself may monitor its components and perform fault mitigation on its own. The plot of components whose errors are handled by a given fault mitigation entity is the fault region. The scope of effect of this fault region shall be defined.

Figure 11 illustrates the scope of fault regions in a simplified SAN example and how they may be recursive in nature. AN HFM application has the widest scope of concern in this example.

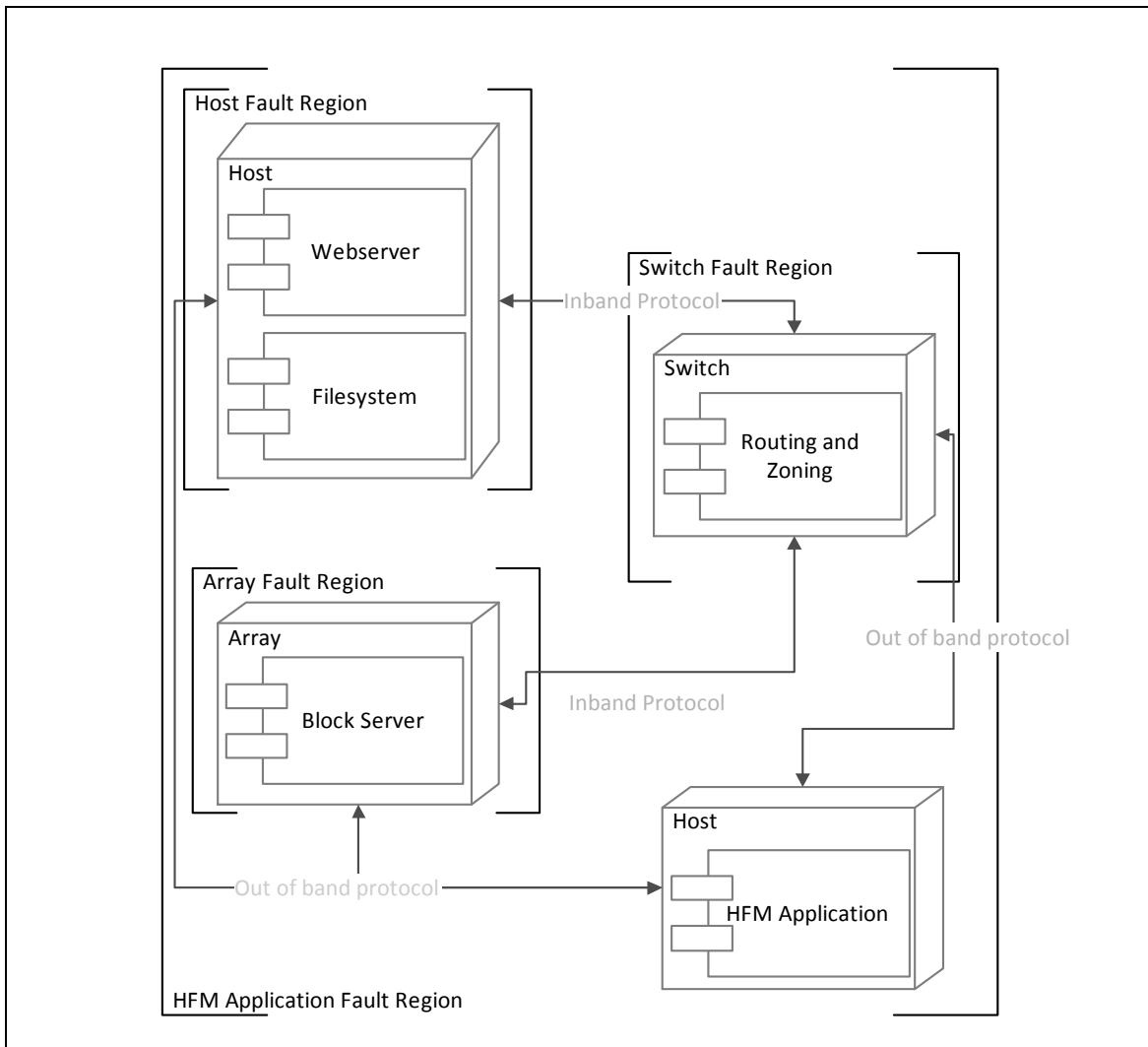


Figure 11 - Application Fault Region

Error handling is initiated by the interception of error events. For example, a switch may recognize the failure of one of its ports and reroute traffic to a working port. In this case, the fault region is defined as the switch itself. If the failure event is publicly consumable, other fault mitigation entities can also handle the error as well. The failure of a drive may be mitigated one way in the array fault region and mitigated differently in the HFM enabled application fault region. For example, the array fault mitigation entity can bring a volume off line if the failure of the disk brings the set of disks below the minimum required for quorum. At the same time, the HFM enabled application can reconfigure the storage service to create a replacement volume and then restore the failed volume's data from backup.

The HFM enabled application is one of the several possible storage network scope fault mitigation entities. As discussed previously, this broad scope is necessary to mitigate faults where the faults cannot be entirely mitigated by the storage device or application alone. It is necessary that fault mitigation entities like the HFM enabled application can observe the activities of the fault mitigation entities contained within their fault regions such that they do no harm. Device or application should express what error conditions are to be handled inside their own fault domain and how an HFM enabled application can detect that such fault containment is occurring. State changes on components may be sufficient representation of these activities.

Array Example

In general, the HFM enabled application fault region mitigation may not necessarily include the same actions that the host, switch, or array may take to fix them.

EXPERIMENTAL

5.4.6 Examples

5.4.6.1 Array Example

The scenario presented is related to a storage array that contains one or many ports. A port is off-line. This port effects access to a volume from a host

An indication is produced by the array notifying the HFM enabled application of the failure. The indication reports the name of the port instance that has failed.

A client that gets the top-level ComputerSystem instance should see an operational status of Degraded. At the same time, reading the instance of the LogicalPort subclass (e.g., FCPort) representing the failed port would see an operational status of Stopped.

5.4.6.2 Switch Example

The scenario presented is related to a FC Switch that contains many ports. See Figure 12. One of the ports is off-line.

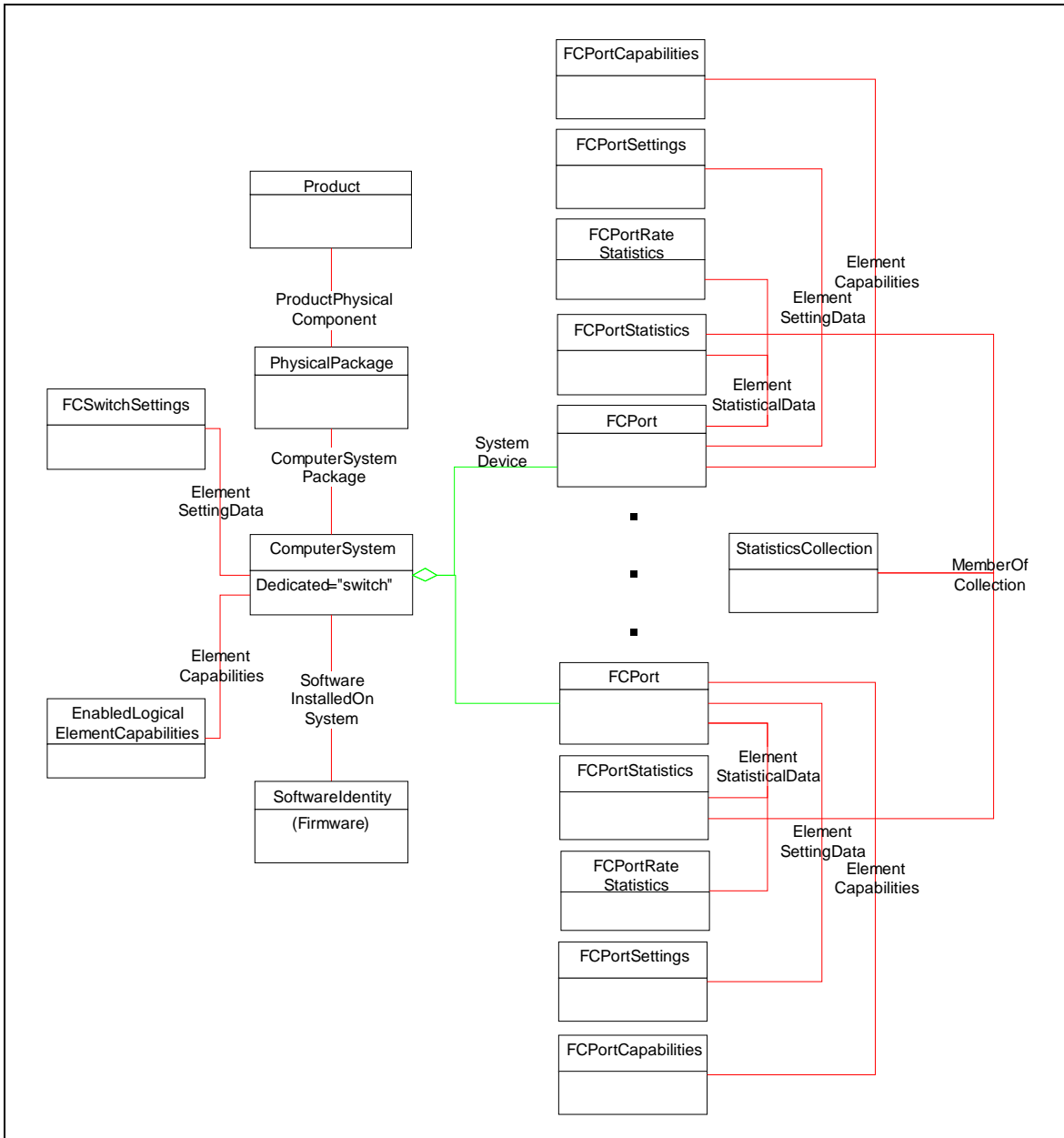


Figure 12 - Switch Example

5.4.6.2.1 Indication

An AlertIndication is produced by the switch notifying the HFM enabled client of the failure. The indication reports the Object Name of the FC port (FCPort) that has failed through its AlertingManagedElement property.

5.4.6.2.2 Standard Errors

A call to Port settings, port capabilities, or statistics cause an Error to be reported. The error reports the Object Name of the FCPort that has failed through the ErrorSource property.

Switch Example

It is mandatory to report error conditions through both AlertIndication 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 heat, 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

5.4.6.2.3 Fault Region

The RelatedElementCausingError association defines the relationship between a CIM Instance that is reporting an error status and the component that is the cause of the reported status. The failed port would report error status and the RelatedElementCausingError association reports that the PortStatistics and PortSettings are affected. The switch itself would be thereby degraded.

The _RelatedElementCausingError association is independent of all other associations. It is only use to report error associations and comes into existence only when necessary. Once the error has been handled, the association is removed from the model.

EXPERIMENTAL

6 Object Model General Information

6.1 Model Overview (Key Resources)

6.1.1 Overview

The SMI-S object model is based on the Common Information Model (CIM), developed by the DMTF. For a more complete discussion of the full functionality of CIM and its modeling approach, see http://www.dmtf.org/standards/standard_cim.php.

Readers seeking a more complete understanding of the assumptions, standards and tools that assisted in the creation of the SMI-S object model are encouraged to review the following:

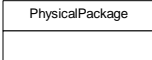
- CIM Tutorial (<http://www.wbemsolutions.com/tutorials/CIM/index.html>)
- CIM UML Diagrams and MOFs (http://www.dmtf.org/standards/standard_cim.php)

Managed Object File (MOF) is a way to describe CIM object definitions in a textual form. A MOF can be encoded in either Unicode or UTF-8. A MOF can be used as input into a MOF editor, parser or compiler for use in an application.

The SMI-S model is divided into several *profiles*, each of which describes a particular class of SAN entity (such as disk arrays or FibreChannel Switches). These profiles allow for differences in implementations but provide a consistent approach for clients to discover and manage SAN resources. In DMTF parlance, a *provider* is the instrumentation logic for a profile. In many implementations, providers operate in the context of a *WBEM Server* that is the infrastructure for a collection of providers. A *WBEM client* interacts with one or more WBEM Servers.

6.1.2 Introduction to CIM UML Notation

CIM diagrams use a subset of Unified Modeling Language (UML) notation.

Most classes are depicted in rectangles.  The class name is in the upper part and *properties* (also known as *attributes* or *fields*) are listed in the lower part. A third subdivision added for *methods*, if they are included. A special type of class, called an *association*, is used to describe the relationship between two or more CIM classes

Three types of lines connect classes, as shown in Figure 13.

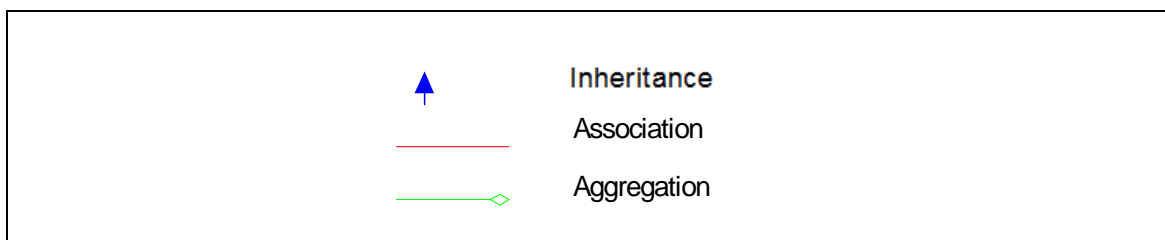


Figure 13 - Lines that Connect Classes

The CIM documents generally follow the convention of using blue arrows for inheritance, red lines for associations and green lines for aggregation. The color-coding makes large diagrams much easier to read but is not a part of the UML standard.

The ends of some associations have numbers (cardinality) indicating the valid count of object instances. Cardinality is expressed either as a single value (such as 1), or a range of values (0..1 or 1..4); "*" is shorthand for 0..n.

Some associations and aggregations are marked with a “W” at one end indicating that the identity of this class depends on the class at the other end of the association. For example, fans may not have worldwide unique identifiers; they are typically identified relative to a chassis.

This document uses two other UML conventions.



The UML Package symbol is used as a shortcut representing a group of classes that work together as an entity. For example, several classes model different aspects of a disk drive. After the initial explanation of these objects, a single disk package symbol is used to represent the entire group of objects.

Schema diagrams include all of a profile’s classes and associations; the class hierarchy is included and each class is depicted one time in the schema diagram. Instance diagrams also contain classes and associations but represent a particular configuration; multiple instances of an object may be depicted in an instance diagram. An instance may be named with an instance name followed by a colon and a class name (underlined). For example,



represents an array and a switch – two instances of <COMPUTER SYSTEM> objects.

6.2 Techniques

6.2.1 CIM Fundamentals

This section provides a rudimentary introduction to some of the modeling techniques used in CIM, and is intended to speed understanding of the SMI-S object model.

6.2.1.1 Associations as Classes

CIM presents relationships between objects with specialized classes called *associations* and *aggregations*. In addition to references to the related objects, the association or aggregations may also contain domain-related properties. For example, *ControlledBy* associates a controller and a device. There is a many-to-many cardinality between controllers and devices (i.e., a controller may control multiple devices and multi-path devices connect to multiple controllers); each controller/device connection has a separate activity state. This state corresponds to the *AccessState* property of *ControlledBy* association linking the device and the controller.

6.2.1.2 Logical and Physical Views

CIM separates physical and logical views of a system component, and represents them as different objects – the “realizes” association ties these logical and physical objects together.

6.2.1.3 Identity

Different agents may each have information about the same organic object and may need to instantiate different model objects representing the same thing. Access control is one example: a switch zone defines which host device ports may access a device port. The switch agent creates partially populated port objects that are also created by the HBA and storage system agents. The *ConcreteIdentity* association is used to indicate the associated object instances are the same thing. *ConcreteIdentity* is also used as a language-independent alternative to multiple inheritance. For example, a *FibreChannel* port inherits from a generic port and also has properties of a *SCSI* controller. CIM models this as *FCPort* and *ProtocolController* objects associated by *ConcreteIdentity*.

6.2.1.4 Extensibility

CIM makes allowances for additional values in enumerations that were not specified in the class Derivation by adding a property to hold arbitrary additional values for an enumeration. This property is usually named OtherXXXX (where XXXX is the name of the enumeration property) and specifying "other" as the value in the enumeration property indicates its use. For an example see the ConnectorType and OtherTypeDescription properties of Slot object in the CIM_Physical MOF.

6.2.1.5 Value/ValueMap Arrays

CIM uses a pair of arrays to represent enumerated types. ValueMap is an array of integers; Values is an array of strings that map to the equivalent entry in ValueMap. For example, PrinterStatus (in the CIM_Device MOF) is defined as follows:

```
ValueMap {"1", "2", "3", "4", "5", "6", "7"},
Values {"Other", "Unknown", "Idle", "Printing", "Warm-up",
"Stopped Printing", "Offline"},
```

A status value of 6 means "Stopped Printing". A client application can automatically convert the integer status value to a human-readable message using this information from the MOF.

6.2.1.6 Return Codes

When a class definition includes a method, the MOF includes Value/ValueMap arrays representing the possible return codes. These values are partitioned into ranges of values; values from 0 to 0x1000 are used for return codes that may be common to various methods. Interoperable values that are specific to a method start at 0x1001; and vendor-specific values may be defined starting at 0x8000. Here's an example of return codes for starting a storage volume.

```
ValueMap {"0", "1", "2", "4", "5", ".", "0x1000",
"0x1001", "...", "0x8000.."},
Values {"Success", "Not Supported", "Unknown", "Time-out",
"Failed", "Invalid Parameter", "DMTF_Reserved",
"Method parameters checked - job started",
"Size not supported",
"Method_Reserved", "Vendor_Specific"}]
```

6.2.1.7 Model Conventions

This is a summary of objects and associations that are common to multiple profiles.

PhysicalPackage represents the physical storage product. PhysicalPackage may be sub-classed to ChangerDevice, but PhysicalPackage accommodates products deployed in multiple chassis.

Product models asset information including vendor and product names. Product is associated with PhysicalPackage.

SoftwareIdentity models firmware and optional software packages. InstalledSoftwareIdentity associates SoftwareIdentity and ComputerSystem, ElementSoftwareIdentity associates SoftwareIdentity and LogicalDevices (a superclass of devices and ports).

Service models a configuration interface (for example, a switch zoning service or an array access control service). Services typically have methods and properties describing the capabilities of the service. A storage system may have multiple services; for example, an array may have separate services for LUN Masking and LUN creation. A client can test for the existence of a named service to see if the agent is providing this capability.

LogicalDevice (for example, FCPort) is a superclass with device subclasses (like and DiskDrive and TapeDrive) and also intermediate nodes like Controller and FCPort. Each LogicalDevice subclass shall

be associated to a ComputerSystem with a SystemDevice aggregation. Due to the large number of LogicalDevice subclasses, SystemDevice aggregations are often omitted in instance diagrams in this specification.

This specification covers many common storage models and management interfaces, but some implementations include other objects and associations not detailed in the specification. In some cases, these are modeled by CIM schema elements not covered by this document. When vendor-specific capabilities are needed, they should be modeled in subclasses of CIM objects. These subclasses may contain vendor-specific properties and methods and vendor-specific associations to other classes.

6.2.2 Modeling Profiles

In addition to modeling SAN components, SMI-S servers shall model the profiles they provide. This information is used two ways:

- Clients can quickly determine which profiles are available.
- An SLP component can query the SMI-S Server and automatically determine the appropriate SLP Service Template information (see Clause 9 Service Discovery).

A client can traverse the Server Profile in each SMI-S server to see which profiles (and objects) claim SMI-S compliance. RegisteredProfile describes the profiles that a WBEM server claims are supported. The RegisteredSubprofile is used to define the optional features supported by the system being modeled. A client can traverse the associations in the Server Profile to see which profiles claim SMI-S compliance.

6.2.3 CIM Naming

There may be multiple SMI-S servers in any given storage network environment. It is not sufficient to think of the name of an object as just the combination of its key values. The name also serves to identify the Server that is responsible for the object. The name of an object (instance) consists of the namespace and the model. The namespace provides access to a specific SMI-S server implementation and is used to locate a particular namespace within a server. The model provides full navigation within the CIM Schema and is the concatenation of the class name and key-qualified properties and values.

The namespace has special rules. It should uniquely identify a SMI-S server. However, a SMI-S server may support multiple namespaces. How an implementation defines Namespaces within a SMI-S server is not restricted. However, to ease interoperability, SMI-S implementations should manage all objects within a profile in one namespace.

7 Correlatable and Durable Names

7.1 Overview

Management applications often read and write information about managed objects in multiple CIM namespaces or between CIM and some other storage management namespace. When an object in one namespace is associated with an object in another namespace, each namespace may represent some amount of information about the same managed resource using different objects. A management application understands when objects in different namespaces represent the same managed resource by the use of a unique common identifier, referred to as a “correlatable name”. A correlatable name is designated as a mandatory property for any objects representing managed resources that may be seen from multiple points of view. These durable names are used by management applications for object coordination.

A related concept is referred to as “durability”. Some names may be correlatable at a particular point in time, but may change over time (e.g., a durable name is a hardware-assigned port or volume name and a correlatable, non-durable ID is a DHCP IP address). No name is permanently durable (e.g., even a name derived from hardware may change due to FRU replacement). A client application should assume that a stored durable name remains valid over time where a non-durable may not remain valid over time.

Correlatable names are unique within a defined namespace. In some cases, that namespace is world-wide; requiring compliance to standards defined by a naming authority. In other cases, the namespace is the hosting system or some set of connected systems (e.g., operating system device names are unique to the containing host).

A name may be expressed in different formats (e.g., numeric value are sometimes displayed as decimal or hexadecimal, the hexadecimal value sometimes has a leading “0x” or a trailing “h”). To assure interoperability, mandatory formats are specified by this standard.

A necessary technique associated with correlatable names involves the use of CIM properties that describe the format or namespace from which the name is derived. CIM key-value combinations are unique across instances of a class, but CIM does not fully address cases where different types of identifiers are possible on different instances of an object. It is therefore necessary to ensure that multiple sources of information about managed resources use the same approach for forming correlatable names whenever different types of identifiers are possible.

When different types of identifiers are possible, the profile specifies the possible name formats and namespaces for durable and correlatable IDS, the preferred order that each implementation should use if multiple namespaces are available, and the related properties that a client uses to determine the namespace.

Correlatable, durable names are mandatory for these objects:

- SCSI logical units or (such as storage volumes or tape drives) that are exported from storage systems; also SB (Single Byte Command Code Sets)
- SB control unit issues
- External Ports on hosts and storage devices
- Fibre Channel ports on interconnect elements
- Fibre Channel fabric (modeled as AdminDomain)
- ComputerSystem objects that server as top-level systems for all SMI-S profiles
- Operating System Device Names

CIM keys and correlatable names are not tightly coupled. For some classes, they may be the same, but this is not mandatory as long as all correlatable names are unique and management applications are able to determine when objects in different namespaces are providing information about the same managed resource.

The common types of information used for names include the SCSI Device Identifiers from the Identification Vital Product Data page (i.e., VPD page 83h), SB Node Element Descriptors from Read-Configuration Data, the response from ATA IDENTIFY commands, Fibre Channel Name_Identifiers (i.e., World Wide Names), Fully Qualified Domain Names, and IP Address information. See 7.2, 7.3, 7.4, and 7.5 for general information on the advantages and disadvantages of certain types of names. The details for each class requiring durable correlatable names are provided in the profiles subclauses of this document.

If the name used in the instrumentation is binary, the CIM representation is an upper case hexadecimal-encoded representation of the value returned. For example, decimal 27 is hexadecimal 1b and will be represented by the string "1B". Note that each binary byte requires two ASCII characters using this representation. If the name used in the instrumentation is ASCII text, the case of the characters is preserved in the CIM property.

7.2 Guidelines for SCSI Logical Unit Names

The preferred logical unit identifier is returned from a SCSI INQUIRY command in VPD page 83h.

NOTE Legacy systems may lack correlatable names as SCSI standards prior to SAM-3 and SPC-3 did not clearly define logical unit names, however this has been clarified to be logical unit names and recent systems have converged in compliance.

The Unit Serial Number VPD page (i.e., SCSI Inquiry VPD Page 80h) returns a serial number, but the SPC-3 standard allows this either be a serial number for a single logical unit or a serial number of the target device. There's no mechanism to discover which approach the device is using. If a client is not coded to understand which products provide per-logical unit or per-target serial numbers, then it should not use the Unit Serial Number VPD page as a logical unit name.

The Identification Vital Product Data page (i.e., VPD page 83h) returns a list of identifiers with metadata describing each identifier. The metadata includes:

- Code Set (binary versus ASCII)
- Association (indicates the SCSI object to which the identifier applies, e.g., for a logical unit, port, or target device)
- Type (the naming authority for identifiers of the structure of information about target ports)
- Protocol Identifier (indicates the SCSI transport protocol to which the identifier applies)

To identify a logical unit name the Association shall be set to zero. The preferred Types for logical units are 3 (NAA), 2 (EUI), and 8 (SCSI Name). However type 1 (T10) is allowed. If the code set in the inquiry response indicates the identifier is binary, the CIM representation is hexadecimal-encoded.

7.3 Guidelines for FC-SB-2 Device Names

FC-SB-2 devices and control unit images use the node-element descriptor (NED) name format. NEDs are retrieved within a configuration record retrieved by the READ-CONFIGURATION DATA command. A configuration record contains information describes the internal configuration of the device, where the information retrieved describes the corresponding node elements that are accessed when an I/O operation is performed.

NEDs are 32 bytes and contain these fields:

- 4 bytes (flags, type, class, reserved) - binary
- 6 byte "type number" - string
- 3 byte "model number" - string
- 3 byte "manufacturer" - string
- 2 byte "plant of manufacture"- string
- 12 byte sequence number" - string
- 2 byte tag - binary

The I/O-Device NED is used for identifying devices. The Token NED is used for identifying control-unit images.

The Name property for LogicalDevices representing SB devices is world-wide unique value formed by composing these fields.

7.4 Guidelines for Port Names

The following is a list of optimal names for ports based on the transport type:

- 1) Fibre Channel ports use Port World Wide Names (i.e., FC Name_Identifier)
- 2) iSCSI has three types of ports
 - the combination of IP address and TCP port number serve as the primary correlatable name for iSCSI target ports. Note that this information is stored in two separate properties and hence there is no single correlatable name.
 - the logical element (iSCSIProtocolEndpoint) that represents the SCSI port The SCSI logical port shall be named with an iSCSI name.
 - the underlying physical ports (typically Ethernet ports). Ethernet ports names shall use the MAC address.
- 3) Parallel SCSI (SPI) and ATA ports typically do not have names, they are identified by a bus-relative address typically set with jumpers. In configurations where these drives are not shared by multiple hosts, the host-relative name acts as the name.
- 4) CIM port classes do not include NameFormat; the appropriate format is determined by the transport implied by the port subclass.

SCSIProtocolEndpoint represents SCSI protocol running through a port. In many cases, there is one-to-one mapping between SCSIProtocolEndpoint and some subclass of LogicalPort and the name requirements are identical. For iSCSI, there may be multiple Ethernet ports per SCSIProtocolEndpoint instance. The IP address and TCP port number are modeled in IPProtocolEndpoint and TCPProtocolEndpoint. iSCSIProtocolEndpoint Name holds the iSCSI initiator or target name.

SBProtocolEndpoint represents SB protocol running through a port. In many cases, there is a one to-one mapping between SBProtocolEndpoint and some subclass of LogicalPort and the name requirements are identical.

7.5 Guidelines for Storage System Names

Each profile has a ComputerSystem or AdminDomain instance that represents the entire system. There are a variety of standard and proprietary names used to name storage systems. Unlike SCSI logical units

and ports, there is no particular name format in common use. There are advantages and disadvantages to certain types of names.

IP addresses have an advantage in human recognition; (e.g., administrators are accustomed to referring to systems by their IP addresses). The downsides are that IP addresses are not necessarily durable (e.g., DHCP) are not necessarily system-wide (e.g., some storage systems have multiple network interfaces), and are not necessarily unique (e.g., NAT allows the same IP address to be used in multiple network zones).

Full Qualified Domain Names are friendlier than IP addresses and may fix the durability issue of IP addresses (e.g., a host name may be constant even when the IP address changes). But storage systems do not necessarily have access to their network names. Network names are typically handled through a central service such as DNS. When a client application opens a connection to a remote system, it asks the local system to resolve the name to an IP address, the local system redirects the request to the DNS server, the IP address is returned and the client application opens the connection. If the remote system is the storage system, this sequence requires the DNS server to know about the storage system, but not vice-versa. A storage system is only required to know about DNS if software on the storage system acts as a network client using host names. And, like IP addresses, a storage system may have several network interfaces with different FQDNs.

Transport-specific names are specific to a particular storage transport (e.g., Fibre Channel or iSCSI). There are some good standard names (e.g., FC platform names or iSCSI Network Entity names). The disadvantage of transport-specific names is that they are not able to be consistently used on storage systems supporting multiple transports or in configurations with transport bridges (e.g., a client may have no mechanism to issue FC commands to an FC device behind an FC/iSCSI bridge).

SCSI target names solve the transport-specific issue. Before the SAM-3 and SPC-3 standards there was not a standard SCSI system name, however with SPC-3, the Identification Vital Product Data page association value 2 was defined for a target name. At this time, the SPC-3 standard is too new to be in common use. Most storage systems include some vendor-specific way to get a target name, but client is not able to use these names without specific knowledge of the vendor-specific interface.

At this time, no single storage system name format is in common use. The best approach is for implementations to expose several names, along with information that tells the client how to interpret the name. The OtherIdentifyingInfo and IdentifyingDescriptions array properties of ComputerSystem provide the list of names and interpretations. However, IdentifyingDescriptions is not an enumerated type; and as a result, any string is valid from a CIM perspective.

7.6 Standard Formats for Correlatable Names

7.6.1 General

Correlatable names shall be used and formatted consistently. Storage volume names are more complex than other element names (i.e., the same format may be used in different namespaces). For example several common INQUIRY Vital Product Data page names use the IEEE NAA format and as a result a client is not able to correlate names from different namespaces.

7.6.2 Standard Formats for Logical Unit Names

For disks and arrays, multiple name formats are in common use. Table 2 specifies standard formats for storage volume names.

Table 2 - Standard Formats for StorageVolume Names

Description	Format property and value(valuemap)	Format of Name
SCSI VPD page 83 type 3, Association 0, NAA 0101b	NameFormat = NAA(9), NameNamespace = VPD83Type3(1)	NAA name with first nibble of 5. Recommended format (8 bytes long) when the ID is directly associated with a hardware component. Formatted as 16 un-separated upper case hex digits (e.g., '21000020372D3C73')
SCSI VPD page 83, type 3h, Association=0, NAA 0110b	NameFormat = NAA(9), NameNamespace= VPD83Type3(1)	NAA name with first nibble of 6. Recommended format (16 bytes long) when IDs are generated dynamically. Formatted as 32 un-separated upper case hex digits.
SCSI VPD page 83, type 3h, Association=0, NAA 0010b	NameFormat = NAA(9), NameNamespace = VPD83Type3(1)	NAA name with first nibble of 2. Formatted as 16 un-separated upper case hex digits
SCSI VPD page 83, type 3h, Association=0, NAA 0001b	NameFormat = NAA(9), NameNamespace = VPD83Type3(2)	NAA name with first nibble of 1. Formatted as 16 un-separated upper case hex digits
SCSI VPD page 83, type 2h, Association=0	NameFormat = EUI64(10), NameNamespace = VPD83Type2(3)	Formatted as 16, 24, or 32 un-separated upper case hex digits
SCSI VPD page 83, type 1h, Association=0	NameFormat = T10VID(11), NameNamespace = VPD83Type1(4)	Formatted as 1 to 252 bytes of ASCII.
SCSI VPD page 80, serial number	NameFormat = Other(1), NameNamespace = VPD80(5)	Only if serial number refers to logical units rather than the enclosure. 1-252 ASCII characters
SB I/O Device NED	NameFormat=SBDevice(13), NameNamespace=SB	64 un-separated upper case hex digits. The tag subfield contains CU_image+device_address
SB Token NED	NameFormat=SBToken(14), NameNamespace=SB	64 un-separated upper case hex digits. The tag sub-field contains the CU_image
SCSI Concatenation of Vendor,Product, SerialNumber	NameFormat = SNVM(7), NameNamespace = SNVM(7)	A concatenation of three strings representing the vendor name, product name within the vendor namespace, and serial number within the model namespace. These strings come from SCSI standard INQUIRY response data. Strings are delimited with a '+' and spaces are included. Vendor and Product are fixed length: Vendor ID is 8 bytes, Product is 16 bytes. SerialNumber is variable length and may be up to 252 bytes in length. If one of these fields contains a plus sign, it shall be escaped with a backslash ('\+'). The concatenation is done to provide world-wide uniqueness; clients should not parse this name.

Table 2 - Standard Formats for StorageVolume Names

Description	Format property and value(valuemap)	Format of Name
ATA Concatenation of, Model, SerialNumber	NameFormat=ATA, NameNamespace=ATA	A concatenation of three strings representing the vendor and model names and serial number within the model namespace. The manufacturer name is not based on a specific standard. The model name and serial number strings come from ATA IDENTIFY DEVICE response data. Strings are delimited with a '+' and spaces are included. The vendor is 20 characters, model is 40 characters, and serial number is 20 characters. If one of these fields contains a plus sign, it shall be escaped with a backslash ('\+'). The concatenation is done to provide uniqueness; clients should not parse this name. Note that ATA standards do not require any interface to return a manufacturer ID; many implementations put a manufacturer name in the model string.
FC Node WWN	NameFormat = NodeWWN(8) NameNamespace = NodeWWN(6)	16 un-separated upper case hex digits (e.g., '21000020372D3C73')

Storage volumes may have multiple standard names. A page 83 logical unit identifier shall be placed in the Name property with NameFormat and Namespace set as specified in Table 2. Each additional name should be placed in an element of OtherIdentifyingInfo and the corresponding element in IdentifyingDescriptions shall contain a string from the Values lists from NameFormat and NameNamespace, separated by a semi-colon. For example, an identifier from SCSI VPD page 83 with type 3, association 0, and NAA 0101b - the corresponding entry in IdentifyingDescriptions[] shall be "NAA;VPD83Type3".

For other types of devices, the logical unit name shall be in the Name property; NameFormat and NameNamespace are not valid properties of these other device classes.

7.6.3 Standard Formats for Port Names

Table 3 specifies standard formats for port names.

Table 3 - Standard Formats for Port Names

An IP interface's MAC	Network Port Permanent Address property; no corresponding format property	Six upper case hex bytes, bytes are delimited by colons ':'
World Wide Name (i.e., FC Name_Identifier)	FCPort Permanent Address property; no corresponding format property	16 un-separated upper case hex digits (e.g., '21000020372D3C73')
	ProtocolEndpoint Name property; ConnectionType = 2 (Fibre Channel)	16 un-separated upper case hex digits (e.g., '21000020372D3C73')
Parallel SCSI Name	SPI Port Name property; no corresponding format property	String - platform-specific name representing the name. Note that this name is only correlatable relative to the system containing the port.
	SCSIProtocolEndpoint Name property; ConnectionType = 3 (Parallel SCSI)	String - platform-specific name representing the name.

Table 3 - Standard Formats for Port Names

An IP interface's MAC	Network Port Permanent Address property; no corresponding format property	Six upper case hex bytes, bytes are delimited by colons ':'
iSCSI Port Name	iSCSIProtocolEndpoint Name	< iSCSI node name > + ' i, ' + ISID for initiators, < iSCSI node name > + ' t, ' + TPGT for target ports, where < iSCSI node name > may be any of the standard iSCSI name namespaces (e.g., iqn, eui); and includes the namespace prefix.
SAS Port Names	SASPort Name property; no corresponding format property	SAS Address, 16 un-separated upper case hex digits
	SCSIProtocolEndpoint Name property; ConnectionType = 8 (SAS)	SAS Address, 16 un-separated upper case hex digits
ATA Port Name	ATAPort or SASSATAPort Name property; no corresponding format property	String - platform-specific name representing the name. Note that this name is only correlatable relative to the system containing the port.
	ATAProtocolEndpoint Nameproperty	String - platform-specific name representing the name.

Note that iSCSI Network Portals do not have a single correlatable name. The combination of IPProtocolEndpoint IPv4Address or IPv6Address and TCPProtocolEndpoint PortNumber uniquely identifies the network portal, but since these are two properties, they do not form a correlatable name.

7.6.4 Standard Formats for Fabric Names

A fabric is modeled as AdminDomain. AdminDomain.Name shall hold the fabric name (i.e., WWN) and AdminDomain.NameFormat shall be set to "WWN". AdminDomain.Name shall be formatted as 16 unseparated upper case hex digits.

7.6.5 Standard Formats for Storage System Names

Due to the limited list of possible formats, the Name property is not considered an essential identifier for SMI-S. SMI-S clients should use OtherIdentifyingInfo property as described in Table 4.

Providers shall supply at least one Durable or Correlatable Name as an element in the IdentifyingDescriptions[] array. The corresponding array elements of OtherIdentifyingInfo[] shall include a value from Table 4 for all elements of IdentifyingDescriptions[]. The elements in the IdentifyingDescriptions array are strings and may contain white space between words. Whenever white-space appears, it shall consist of a single blank; other white-space characters and multiple consecutive blanks shall not be used.

At least one of the values in IdentifyingDescriptions[] shall be something other than "SCSI Vendor Specific Name" or "Other Vendor Specific Name".

OtherIdentifyingInfo[0] should be assigned the most preferable name by the instrumentation.

In all cases, if the name is returned to the instrumentation in binary, the corresponding entry in OtherIdentifyingInfo holds an upper-case hexadecimal-encoded representation of the value returned. Standard names defined in binary are called out in Table 4.

Other ComputerSystem properties should be set as follows:

Name is a CIM key and shall be unique for ComputerSystem instances within the CIM namespace. SMI-S clients should not assume Name is either durable or correlatable.

NameFormat is an enumerated type describing the Name property. Only a few of the defined values are appropriate for storage systems. Use "IP" if Name is derived from an IP address of Fully Qualified Domain Name. Use "HID" if Name is derived from a hardware ID. Use "OID" if Name is a unique ID determined by some unique ID generating logic.

ElementName is a friendly name; SMI-S clients should not assume that ElementName is unique, correlatable, or durable since a customer may provide the same info for multiple systems.

Table 4 - Standard Formats for Storage System Names

IdentifyingDescriptions [x] value	Description		Format of OtherIdentifyinginfo[x]
T10 Target Name Type 1	An identifier from a Identification Vital Product Data page response with Association equal to 2	Type 1 (T10)	1 to 252 bytes of ASCII
T10 Target Name Type 2		Type 2 (EUI)	16, 24, or 32 un-separated upper case hex digits (e.g., '21000020372D3C73')
T10 Target Name Type 3		Type 3 (NAA)	16 or 32 un-separated upper case hex digits (e.g., '21000020372D3C73')
T10 Target Name Type 8		Type 8 (SCSI Names)	iSCSI Names (see 7.8)
T11 FC-GS-4 Platform Name	A platform name as defined in T11 FC-GS-4 standard		Up to 508 hex digits (254 bytes) as specified by T11 FC-GS-4 subclause on Platform Name. Format as un-separated as hex digits. Platform Name Format Byte shall be included.
T11 RNID Name	The sixteen byte Vendor Unique name from the General Topology Discovery format RNID response as defined in T11 FC LS standard. This name format should only be used if the storage system supports RNID General Topology Discovery and provides a meaning system identifier in the Vendor Unique field.		32 un-separated hex digits.
iSCSI Network Entity Name	An iSCSI Network Entity name.		iSCSI Names (see 7.8)
Ipv4 Address	An IP V4 name		Four decimal bytes delimited with dots ('.')
Ipv6 Address	An IP V6 name		'x:x:x:x:x:x:x', where the 'x's are the uppercase hexadecimal values of the eight 16-bit pieces of the address. Examples: 'FEDC:BA98:7654:3210:FEDC:BA98:7654:3210', '1080:0:0:0:8:800:200C:417A' Leading zeros in individual fields should not be included and there shall be at least one numeral in every field. (This format is compliant with RFC 4291.) In addition, omitting groups of zeros or using dotted decimal format for an embedded IPv4 address is prohibited.
Fully Qualified Domain Name	A fully qualified domain name.		A legal DNS name (fully qualified) consisting of strings delimited by periods.

Table 4 - Standard Formats for Storage System Names (Continued)

Identifying Descriptions [x] value	Description		Format of Other Identifying info[x]
Node WWN	The Fibre Channel Node WWN. The provider shall assure that the same Node WWN shall be available through all FC ports within a target device.		16 un-separated upper case hex digits (e.g., '21000020372D3C73')
T10 Unit Serial Number VPD page	SCSI Inquiry VPD page 80 response is a serial number This name may be unique for a specific logical unit or for the target (e.g., storage system). These names are only valid if the instrumentation is certain that all logical units in a system return the same value. Since there is no mechanism to test whether the value is unique per target or per logical unit, this value is not interoperably correlatable and should not be used		1-252 ASCII characters
SCSI Vendor Specific Name	This is a name accessible through a vendor-specific SCSI command	A client with a priori knowledge may be able to correlate this based on vendor and Product IDs.	unknown
Other Vendor Specific Name	This is a name accessible through some non-SCSI vendor-specific interface.		unknown

7.6.6 Operating System Device Names

Each operating system has different conventions for naming devices. Many operating systems provide multiple names for the same device instance. In this version of the specification, operating system device name formats are recommended.

The case of names specified by operating system interfaces shall be preserved.

Operating system device names are unique within the namespace of the scoping system and are not unique between systems.

Table 5 specifies the format for names of tape devices.

Table 5 - Standard Operating System Names for Tape Devices

Operating System	Format	Notes
AIX	/dev/rmtX	X represents a hexadecimal number and may be more than one character
HP-UX	/dev/rmn/Xm	X represents a hexadecimal number and may be more than one character
Linux	/dev/stX	X represents one or two lower case alphabetic characters
Solaris	/dev/rmt/Xn	X represents a hexadecimal number and may be more than one character
Windows	\\.\TAPEX	X represents a decimal number

Some operating systems treat disk partitions as virtual devices; applications operate on partitions as if they were disks. The model requires two classes for each partition, LogicalDisk and GenericDiskPartition. Other operating systems allow applications to operate on the entire disk without partitions. Linux allows both.

Table 6 specifies the format for LogicalDisk.Name of disk partitions

Table 6 - LogicalDisk.Name for disk partitions

Operating System	Format	Notes
Linux	dev/sdXY or /dev/hdXY	where X represents one or two lower case alphabetic characters and Y represents an integer between 1 and 15
Solaris	/dev/dsk/cXtXdXsX	X represents one or two lower case alphabetic characters
Windows	C: or the file name of mount point	C represents an uppercase letter
zSeries	CC:SS:DDDD or CC:DDDD	CC represents a Channel Subsystem Identifier, SS is a subchannel set (within the channel subsystem), and DDDD is the device number. SS is optional for subchannel set zero.

Table 7 specifies the format for GenericDiskPartition.Name and DeviceId properties for disk partitions

Table 7 - GenericDiskPartition.Name for disk partitions

Operating System	Format	Notes
Linux	sdXY or hdXY	X represents one or two lower case alphabetic characters
Solaris	/dev/dsk/cXtXdXsX	where X represents one or two lower case alphabetic characters and Y represents an integer between 1 and 15
Windows	Disk #X, Partition #X	X represents a decimal digit

Table 8 specifies the format for LogicalDisk.Name for unpartitioned disks.

Table 8 - Standard Operating System Names for Unpartitioned Disks

Operating System	Format	Notes
AIX	/dev/hdiskX	X represents a hexadecimal number and may be more than one character
HP-UX	/dev/dsk/cXtYdZ	X, Y, and Z represents hexadecimal number and may be more than one character in length
Linux	/dev/sdX or /dev/hdX	X represents one or two lower case alphabetic characters
Windows	\\.PHYSICALDRIVEx	x represents a decimal number and may be more than one character

7.6.7 Case Sensitivity

Names and NameFormats are case sensitive and the cases provided in Table 8 shall be used. If not otherwise specified, uppercase should be used.

7.7 Testing Equality of correlatable Names

The implementation shall only compare objects of the same class or parent class. For objects that do not require the use of additional properties, a simple direct comparison is sufficient, providing the format for the mandatory correlatable name as identified in this section or the specific profile is adhered to.

For objects that do require the use of additional properties (e.g., NameFormat), the correlatable names of objects representing the same entity should compare positively, negatively, or indicate clearly when a comparison is ambiguous:

- If the two objects have the same NameFormat and Name, then they refer to the same resource.
- If the two objects have the same NameFormat and different Names, then they refer to different resources.
- If the two objects have different NameFormats, whether the Names are the same or different, then it is unknown whether they refer to the same resource.

This reduces the possibility that a match is missed by a string equals comparison simply because of an incompatibility of formats rather than non-equality of the data.

7.8 iSCSI Names

The iSCSI standards define three text formats for names that apply to various iSCSI elements. The three formats are: iSCSI qualified name (iqn), IEEE Extended Unique Identifier (eui), and ANSI T10 NAA. The format is included in the name as a three-letter prefix. The three formats are explained in more detail.

The iSCSI qualified name (iqn) format is defined in [iSCSI] and contains (in order):

- 1) 1 - The string "iqn."
- 2) 2 - A date code specifying the year and month in which the organization registered the domain or sub-domain name used as the naming authority string.
- 3) 3 - The organizational naming authority string, which consists of a valid, reversed domain or sub-domain name.

Optionally, a ':', followed by a string of the assigning organization's choosing, which shall make each assigned iSCSI name unique.

Figure 14 contains examples of iSCSI-qualified names that may be generated by "EXAMPLE Storage, Inc."

Type	Organizational Naming	Date	Auth	Subgroup Naming Authority and/or string Defined by Org. or Local Naming Authority
iqn.		2001-04	com.example	diskarrays-sn-a8675309
iqn.		2001-04	com.example	
iqn.		2001-04	com.example	storage.tape1.sys1.xyz
iqn.		2001-04	com.example	storage.disk2.sys1.xyz

Figure 14 - iSCSI Qualified Names (iqn) Examples

The IEEE Registration Authority provides a service for assigning globally unique identifiers [EUI]. The EUI-64 format is used to build a global identifier in other network protocols.

The format is "eui." followed by an EUI-64 identifier. Figure 15 contains an example.

Type	EUI-64 identifier (ASCII-encoded hexadecimal)
eui.	02004567A425678D

Figure 15 - iSCSI EUI Name Example

Type "naa." - Network Address Authority

The ANSI T10 FC-FS standard defines a format for constructing globally unique identifiers [FC-FS] referred to as an Network Address Authority (NAA) format. The iSCSI name format is "naa." followed by an NAA identifier (ASCII-encoded hexadecimal digits).

Figure 16 contains an example of an iSCSI name with a 64-bit NAA value: type NAA identifier (ASCII-encoded hexadecimal).

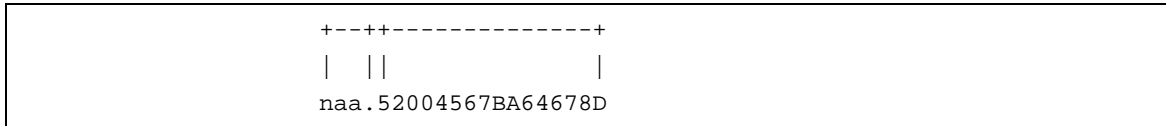


Figure 16 - iSCSI 64-bit NAA Name Example

Figure 17 contains an example of an iSCSI name with a 128-bit NAA value: type NAA identifier (ASCII-encoded hexadecimal).

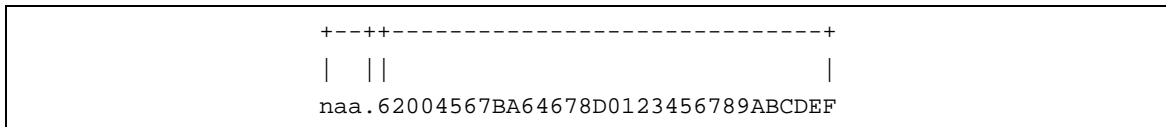


Figure 17 - iSCSI 128-bit NAA Name Example

iSCSI names are composed only of displayable characters. iSCSI names allow the use of international character sets but are not case sensitive. No whitespace characters are used in iSCSI names.

8 Standard Messages

8.1 Overview

Standard Messages provide an interoperable design for event reporting and are documented in *DMTF DSP2011, Standard Messages Whitepaper*. SMI-S profiles reference standard messages for alert indications and error reporting.

8.2 Registries for Standard Messages

Message registries are machine-readable lists of standard messages specified in DMTF DSP0228 1.1.0. Instances of CIM_AlertIndication or CIM_Error classes in implementations of this standard should conform to the provisions specified by the referenced DMTF message registries, or 8.3 "SNIA Standard Messages". Implementations of this standard may utilize the machine-readable capability of message registries, but may use other techniques to assure message payload conforms to provisions of message registries.

CIM_AlertIndication instances referenced in SMI-S profile include OwningEntity and MessageID properties. If OwningEntity is DMTF and the MessageID starts with WIPG, see DSP8016 WBEM Operations Message Registry 1.1. If OwningEntity is DMTF and the MessageID starts with DIAG, see DSP8055 Diagnostics Message Registry. SNIA standard messages are documented in 8.3 "SNIA Standard Messages".

8.3 SNIA Standard Messages

8.3.1 Common Profile-related Messages

8.3.1.1 Message: Redundancy

Owning Entity: SNIA

Message ID: Core1

Message Format String: <Device Type> <Device Unique Identifier> had redundancy failure for <Component Type> at <Component Location Or Identifier>

A message indicating a redundancy failure in a set of redundant components in a device. Table 9 describes the message arguments.

Table 9 - Redundancy Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
Component Type	string	The type of component a redundancy failure has occurred for. Typically the string would contain one of the following: "Power Component", "Fan Component", "Board Component", Cross Bar", "System Clock'	Power Component
			Fan Component
			Board Component

Table 9 - Redundancy Message Arguments

Message Argument	Data Type	Description	Possible Values
			Cross Bar
			System Clock
			Communications Port
Component Location Or Identifier	string	Location or identifier of the component	

Table 10 describes the alerts that are associated with this message.

Table 10 - Redundancy Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system containing the device.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Minor

8.3.1.2 Message: Environmental

Owning Entity: SNIA

Message ID: Core2

Message Format String: <Device Type> <Device Unique Identifier> had an environmental problem of type <SensorType> of <Environmental Issue> <Sensor Location Or Identifier>

A message indicating a environmental issue with a device. Table 11 describes the message arguments.

Table 11 - Environmental Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
			Tape Library
			Drive
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
SensorType	string	The sensor type.	temperature
			humidity
Environmental Issue	string	The environmental issue.	
Sensor Location Or Identifier	string	Location or identifier of the Sensor	

Table 12 describes the alerts that are associated with this message.

Table 12 - Environmental Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		Environmental Alert
PERCEIVED_SEVERITY	Y	4	Minor

EXPERIMENTAL

8.3.1.3 Message: FRU Operation

Owning Entity: SNIA

Message ID: Core3

Message Format String: <Device Type> <Device Unique Identifier> had a Field Replaceable Unit (FRU) <The FRU Operation> on <FRU Type> at <FRU Location Or Identifier>

A message indicating an manual operation occurred with a Field Replaceable Unit (FRU) that resulted in a change. Table 13 describes the message arguments.

Table 13 - FRU Operation Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
The FRU Operation	string	The operation on the FRU that is the basis of the message	removed
			added
			replaced
			incompatible
FRU Type	string	The Type of FRU	
FRU Location Or Identifier	string	Location or the Identifier of the FRU	

Message: Password change

Table 14 describes the alerts that are associated with this message.

Table 14 - FRU Operation Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.1.4 Message: Password change

Owning Entity: SNIA

Message ID: Core4

Message Format String: <Device Type> <Device Unique Identifier> password has change for user <User Identification>

A message indicating a user or account password has change. Table 15 describes the message arguments.

Table 15 - Password change Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
User Identification	string	User or Account Identification	

Table 16 describes the alerts that are associated with this message.

Table 16 - Password change Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.1.5 Message: User or Account Operation

Owning Entity: SNIA

Message ID: Core5

Message Format String: <Device Type> <Device Unique Identifier> user <User Identification> <User Operation>

A message indicating a user or account password has added, removed, or disabled. Table 17 describes the message arguments.

Table 17 - User or Account Operation Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
User Identification	string	User or Account Identification	
User Operation	string	Operation on User	removed
			disabled
			added

Table 18 describes the alerts that are associated with this message.

Table 18 - User or Account Operation Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		Security Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.1.6 Message: User Login

Owning Entity: SNIA

Message ID: Core6

Message Format String: <Device Type> <Device Unique Identifier> user <User> <Login Operation>

Message: Proxy Agent Device Communication

A message indicating user or account login activity including logging into or off of a device. Table 19 describes the message arguments.

Table 19 - User Login Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
User	string	The user or account name	
Login Operation	string	Operation on User	logged in
			logged out

Table 20 describes the alerts that are associated with this message.

Table 20 - User Login Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name must reference the top-most computer system that is shutting down. If the computer system is cluster, then the cluster computer system must be referenced.
ALERT_TYPE	Y		Security Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.1.7 Message: Proxy Agent Device Communication

Owning Entity: SNIA

Message ID: Core7

Message Format String: Agent <Agent Identifier> <Agent Connectivity> communication with <Device Type> <Device Unique Identifier>

If an agent is acting as a proxy to a device, this message is used if the connection is lost between the proxy and the device. Table 21 describes the message arguments.

Table 21 - Proxy Agent Device Communication Message Arguments

Message Argument	Data Type	Description	Possible Values
Agent Identifier	string	An identifier for the SMI Agent	
Agent Connectivity	string	A description for the connectivity	lost
			regained

Table 21 - Proxy Agent Device Communication Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	

Table 22 describes the alerts that are associated with this message.

Table 22 - Proxy Agent Device Communication Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.1.8 Message: Port Status Changed

Owning Entity: SNIA

Message ID: Core8

Message Format String: FC Port <Port Identifier> in <Device Type> <Device Unique Identifier> status changed to <Port Status>

The fabric has detected a change in status of a Fibre Channel port in the fabric. Table 23 describes the message arguments.

Table 23 - Port Status Changed Message Arguments

Message Argument	Data Type	Description	Possible Values
Port Identifier	string	The Fibre Channel Port Name.	
Device Type	string		Switch
			HBA
			Array
			Fabric
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
Port Status	string	Fibre Channel Port Status. This should be the same value as the OperationalStatus for the FCPort.	

Message: Datacheck Error

Table 24 describes the alerts that are associated with this message.

Table 24 - Port Status Changed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.1.9 Message: Datacheck Error

Owning Entity: SNIA

Message ID: Core9

Message Format String: <Device Type> <Device Unique Identifier> data check (<Data Check Type>

A data check error occurred on a device. The error could be a checksum error, CRC error, or some other kind of error where there was some determination that the data transmitted or stored was not correct. Table 25 describes the message arguments.

Table 25 - Datacheck Error Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
Data Check Type	string	The type of data check that occurred.	Switch
			CRC

Table 26 describes the alerts that are associated with this message.

Table 26 - Datacheck Error Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.1.10 Message: User Login Failure

Owning Entity: SNIA

Message ID: Core10

Message Format String: <Device Type> <Device Unique Identifier> user <User> had login failure.

A message indicating user or account login failure into a device. Table 27 describes the message arguments.

Table 27 - User Login Failure Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Type	string	A description of the type of element	Switch
			HBA
			Array
Device Unique Identifier	string	An identifier for the device (host, array, switch, etc.).	
User	string	The user or account name	

Table 28 describes the alerts that are associated with this message.

Table 28 - User Login Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name must reference the top-most computer system that is shutting down. If the computer system is cluster, then the cluster computer system must be referenced.
ALERT_TYPE	Y		Security Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.1.11 Message: Drive not responding

Owning Entity: SNIA

Message ID: Core12

Message Format String: <Type of Drive> drive is not responding. Drive Identifier: <Device Unique Identifier>

Message: Fan Failure

A message indicating a drive is not responding to I/O commands. Table 29 describes the message arguments.

Table 29 - Drive not responding Message Arguments

Message Argument	Data Type	Description	Possible Values
Type of Drive	string	Type of drive not responding.	Disk
			Tape
			CD
			DVD
Device Unique Identifier	string	An identifier for the drive.	

Table 30 describes the alerts that are associated with this message.

Table 30 - Drive not responding Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Reference to the drive
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

8.3.1.12 Message: Fan Failure

Owning Entity: SNIA

Message ID: Core13

Message Format String: Fan failure.

Table 31 describes the alerts that are associated with this message.

Table 31 - Fan Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_Fan (if modelled) or CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.1.13 Message: Power Supply Failure

Owning Entity: SNIA

Message ID: Core14

Message Format String: Power supply unit failure.

Table 32 describes the alerts that are associated with this message.

Table 32 - Power Supply Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_PowerSupply (if modelled) or CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.1.14 Message: Drive Power Consumption

Owning Entity: SNIA

Message ID: Core15

Message Format String: Power consumption of the drive is outside specified range.

Table 33 describes the alerts that are associated with this message.

Table 33 - Drive Power Consumption Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.1.15 Message: Drive Voltage

Owning Entity: SNIA

Message ID: Core17

Message Format String: Drive voltage limits exceeded.

Table 34 describes the alerts that are associated with this message.

Table 34 - Drive Voltage Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.1.16 Message: Predictive Failure

Owning Entity: SNIA

Message ID: Core18

Message Format String: Predictive failure of drive hardware.

Message: Diagnostics Required

Table 35 describes the alerts that are associated with this message.

Table 35 - Predictive Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.1.17 Message: Diagnostics Required

Owning Entity: SNIA

Message ID: Core19

Message Format String: The drive may have a hardware fault that may be identified by extended diagnostics (i.e., SEND DIAGNOSTIC command).

Table 36 describes the alerts that are associated with this message.

Table 36 - Diagnostics Required Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

EXPERIMENTAL

8.3.1.18 Message: Drive is responding

Owning Entity: SNIA

Message ID: Core20

Message Format String: <Type of Drive> drive issues have cleared. Drive Identifier: <Device Unique Identifier>

A message indicating a drive has resumed responding to I/O commands. Table 37 describes the message arguments.

Table 37 - Drive is responding Message Arguments

Message Argument	Data Type	Description	Possible Values
Type of Drive	string	Type of drive responding.	Disk
			Tape
			CD
			DVD
Device Unique Identifier	string	An identifier for the drive.	

Table 38 describes the alerts that are associated with this message.

Table 38 - Drive is responding Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Reference to the drive
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.1.19 Message: Cooling Fan Issues Cleared

Owning Entity: SNIA

Message ID: Core21

Message Format String: Fan issues cleared:

Table 39 describes the alerts that are associated with this message.

Table 39 - Cooling Fan Issues Cleared Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_Fan (if modeled) or CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.1.20 Message: Power Supply Issues Cleared

Owning Entity: SNIA

Message ID: Core22

Message Format String: Power supply unit issues cleared: <Device Unique Identifier>

Table 40 describes the message arguments.

Table 40 - Power Supply Issues Cleared Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Unique Identifier	string	An identifier for the power supply.	

Message: Controller Failure

Table 41 describes the alerts that are associated with this message.

Table 41 - Power Supply Issues Cleared Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_PowerSupply (if modelled) or CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

8.3.1.21 Message: Controller Failure

Owning Entity: SNIA

Message ID: Core23

Message Format String: Controller failure: <Device Unique Identifier>

Table 42 describes the message arguments.

Table 42 - Controller Failure Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Unique Identifier	string	An identifier for the controller.	

Table 43 describes the alerts that are associated with this message.

Table 43 - Controller Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

8.3.1.22 Message: Controller Issues Cleared

Owning Entity: SNIA

Message ID: Core24

Message Format String: Controller issues resolved: <Device Unique Identifier>

Table 44 describes the message arguments.

Table 44 - Controller Issues Cleared Message Arguments

Message Argument	Data Type	Description	Possible Values
Device Unique Identifier	string	An identifier for the controller.	

Message: Device Not ready

Table 45 describes the alerts that are associated with this message.

Table 45 - Controller Issues Cleared Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

8.3.2 Block Storage Messages

EXPERIMENTAL

8.3.2.1 Message: Device Not ready

Owning Entity: SNIA

Message ID: DRM1

Message Format String: Device <Device ID> not ready because of <StatusOrStatus> state or status.

A message indicating a device is not ready. Table 46 describes the message arguments.

Table 46 - Device Not ready Message Arguments

Message Argument	Data Type	Description	Possible Values
Device ID	string	LogicalDevice.DeviceID, PhysicalElement.Tag, or ComputerSystem.Name	
StatusOrStatus	string	Relevant State or Status the most explains the reason for the production of this message.	

Message: Internal Bus Error

Table 47 describes the error properties.

Table 47 - Error Properties for Device Not ready

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	4(Minor)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.2 Message: Internal Bus Error

Owning Entity: SNIA

Message ID: DRM2

Message Format String: Internal Bus Error

Table 48 describes the error properties.

Table 48 - Error Properties for Internal Bus Error

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	4(Minor)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.3 Message: DMA Overflow

Owning Entity: SNIA

Message ID: DRM3

Message Format String: DMA Overflow

Table 49 describes the error properties.

Table 49 - Error Properties for DMA Overflow

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	4(Minor)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.4 Message: Firmware Logic Error

Owning Entity: SNIA

Message ID: DRM4

Message Format String: Firmware Logic Error

Table 50 describes the error properties.

Table 50 - Error Properties for Firmware Logic Error

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	4(Minor)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.5 Message: Front End Port Error

Owning Entity: SNIA

Message ID: DRM5

Message Format String: Front End Port Error on Device identified by <Device ID>

Message: Back End Port Error

A message indicating front end port is in error. Table 51 describes the message arguments.

Table 51 - Front End Port Error Message Arguments

Message Argument	Data Type	Description	Possible Values
Device ID	string	LogicalDevice.DeviceID	

Table 52 describes the alerts that are associated with this message.

Table 52 - Front End Port Error Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Object Name for the top-level object for the device, which is typically the computer system instance
ALERT_TYPE	Y	2	Communications Alert
PERCEIVED_SEVERITY	Y	4	Minor

EXPERIMENTAL

EXPERIMENTAL

8.3.2.6 Message: Back End Port Error

Owning Entity: SNIA

Message ID: DRM6

Message Format String: Back End Port Error on Device identified by <Device ID>

A message indicating a back end port is in error. Table 53 describes the message arguments.

Table 53 - Back End Port Error Message Arguments

Message Argument	Data Type	Description	Possible Values
Device ID	string	LogicalDevice.DeviceID	

Table 54 describes the alerts that are associated with this message.

Table 54 - Back End Port Error Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Object Name for the top-level object for the device, which is typically the computer system instance
ALERT_TYPE	Y	2	Communications Alert
PERCEIVED_SEVERITY	Y	4	Minor

EXPERIMENTAL

EXPERIMENTAL

8.3.2.7 Message: Remote Mirror Error

Owning Entity: SNIA

Message ID: DRM7

Message Format String: Error detected associated with remote volume, <Remote Volume Name>

A message indicating an error associated with remote volume. Table 55 describes the message arguments.

Table 55 - Remote Mirror Error Message Arguments

Message Argument	Data Type	Description	Possible Values
Remote Volume Name	string	StorageVolume.Name	

Table 56 describes the error properties.

Table 56 - Error Properties for Remote Mirror Error

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the remote block server, which is typically the computer system instance. The implementation will have to implement the Cascading Subprofile.	Existence is optional
PERCEIVED_SEVERITY	3(Degraded/Warning)	Existence is required

Table 57 describes the alerts that are associated with this message.

Table 57 - Remote Mirror Error Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	N		Object Name for the top-level object for the remote block server, which is typically the computer system instance. The implementation will have to implement the Cascading Subprofile.
ALERT_TYPE	Y		
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.2.8 Message: Cache Memory Error

Owning Entity: SNIA

Message: Unable to Access Remote Device

Message ID: DRM8

Message Format String: Cache Memory Error

Table 58 describes the error properties.

Table 58 - Error Properties for Cache Memory Error

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	3(Degraded/Warning)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.9 Message: Unable to Access Remote Device

Owning Entity: SNIA

Message ID: DRM9

Message Format String: Unable to Access Remote Device

Table 59 describes the error properties.

Table 59 - Error Properties for Unable to Access Remote Device

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the remote block server, which is typically the computer system instance. The implementation will have to implement the Cascading Subprofile.	Existence is optional
PERCEIVED_SEVERITY	3(Degraded/Warning)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.10 Message: Error Reading Data

Owning Entity: SNIA

Message ID: DRM10

Message Format String: Error Reading Data

Table 60 describes the alerts that are associated with this message.

Table 60 - Error Reading Data Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Object Name for the top-level object for the device, which is typically the computer system instance
ALERT_TYPE	Y	2	Communications Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.2.11 Message: Error Writing Data

Owning Entity: SNIA

Message ID: DRM11

Message Format String: Error Writing Data

Table 61 describes the alerts that are associated with this message.

Table 61 - Error Writing Data Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Object Name for the top-level object for the device, which is typically the computer system instance
ALERT_TYPE	Y	2	Communications Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.2.12 Message: Error Validating Write (CRC)

Owning Entity: SNIA

Message ID: DRM12

Message Format String: Error Validating Write

Message: Copy Operation Failed

Table 62 describes the alerts that are associated with this message.

Table 62 - Error Validating Write (CRC) Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		Object Name for the top-level object for the device, which is typically the computer system instance
ALERT_TYPE	Y	2	Communications Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.2.13 Message: Copy Operation Failed

Owning Entity: SNIA

Message ID: DRM13

Message Format String: Copy Operation Failed

Table 63 describes the error properties.

Table 63 - Error Properties for Copy Operation Failed

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE		Existence is discouraged
PERCEIVED_SEVERITY	3(Degraded/Warning)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.14 Message: RAID Operation Failed

Owning Entity: SNIA

Message ID: DRM14

Message Format String: RAID Operation Failed

Table 64 describes the error properties.

Table 64 - Error Properties for RAID Operation Failed

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	5(Hardware Error)	Existence is required
ERROR_SOURCE		Existence is discouraged
PERCEIVED_SEVERITY	3(Degraded/Warning)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.15 Message: Invalid RAID Type

Owning Entity: SNIA

Message ID: DRM15

Message Format String: Invalid RAID Type

Table 65 describes the error properties.

Table 65 - Error Properties for Invalid RAID Type

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	10(Unsupported Operation Error)	Existence is required
ERROR_SOURCE		Existence is discouraged
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.16 Message: Invalid Storage Element Type

Owning Entity: SNIA

Message ID: DRM16

Message Format String: Invalid Device Type

Message: Configuration Change Failed

Table 66 describes the error properties.

Table 66 - Error Properties for Invalid Storage Element Type

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	10(Unsupported Operation Error)	Existence is required
ERROR_SOURCE		Existence is discouraged
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.17 Message: Configuration Change Failed

Owning Entity: SNIA

Message ID: DRM17

Message Format String: Configuration Change Failed

Table 67 describes the error properties.

Table 67 - Error Properties for Configuration Change Failed

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.18 Message: Buffer Overrun

Owning Entity: SNIA

Message ID: DRM18

Message Format String: Buffer Overrun

Table 68 describes the error properties.

Table 68 - Error Properties for Buffer Overrun

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Object Name for the top-level object for the device, which is typically the computer system instance	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.19 Message: Stolen Capacity

Owning Entity: SNIA

Message ID: DRM19

Message Format String: The capacity requested, <Requested Capacity> , that was requested is no longer available.

A message indicating an invalid capacity was requested. Table 69 describes the message arguments.

Table 69 - Stolen Capacity Message Arguments

Message Argument	Data Type	Description	Possible Values
Requested Capacity	sint64	Capacity requested in bytes expressed in powers of 10.	

Table 70 describes the error properties.

Table 70 - Error Properties for Stolen Capacity

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	The pool, volume, or logical disk being modified, or, in the case of element creation the parent pool from which capacity is being drawn.	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.20 Message: Invalid Extent passed

Owning Entity: SNIA

Message ID: DRM20

Message Format String: One or more of the extents passed can not be used to create or modify storage elements. <Invalid Extents Array>

A message indicating an invalid extent was passed as an argument. Table 71 describes the message arguments.

Table 71 - Invalid Extent passed Message Arguments

Message Argument	Data Type	Description	Possible Values
Invalid Extents Array	reference	Array of references to the all Extents that can not be used in the specified manner (ex. CreateOrModifyStoragePool or CreateOrModifyElementsFromElements).	

Table 72 describes the error properties.

Table 72 - Error Properties for Invalid Extent passed

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	A reference to the storage configuration service instance on which the method was called that caused this error.	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.21 Message: Invalid Deletion Attempted

Owning Entity: SNIA

Message ID: DRM21

Message Format String: Existing pool or storage element (StorageVolume or LogicalDisk) may not be deleted because there are existing Storage Extents which relay on it.

Table 73 describes the error properties.

Table 73 - Error Properties for Invalid Deletion Attempted

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	A reference to one of the dependent StorageExtents.	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.22 Message: Job Failed to Start

Owning Entity: SNIA

Message ID: DRM22

Message Format String: Job failed to start because resources required for method execution are no longer available.

Table 74 describes the error properties.

Table 74 - Error Properties for Job Failed to Start

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	8(Oversubscription Error)	Existence is required
ERROR_SOURCE	Reference to Job instance which failed to start for this reason if a Job instance was created because of the time required to make this resource assessment. If a Job instance was not created, because the assessment was fast enough, then this property must be NULL.	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.23 Message: Job was Halted

Owning Entity: SNIA

Message ID: DRM23

Message Format String: Job was <Reason for Job halt>

Message: Invalid State Transition

A message indicating that a job was halted. Table 75 describes the message arguments.

Table 75 - Job was Halted Message Arguments

Message Argument	Data Type	Description	Possible Values
Reason for Job halt	string	A Job may be stopped by a client using the RequestedStateChange method. If the job stopped executing for other reasons, then use a different message.	killed
			terminated

EXPERIMENTAL

EXPERIMENTAL

8.3.2.24 Message: Invalid State Transition

Owning Entity: SNIA

Message ID: DRM24

Message Format String: An invalid state transition, <Invalid Sync State> , was requested given current state, <Current Sync State>

A message indicating an invalid state transition occurred. Table 76 describes the message arguments.

Table 76 - Invalid State Transition Message Arguments

Message Argument	Data Type	Description	Possible Values
Invalid Sync State	string	The textual equivalent (Value) for StorageSynchronized.SyncState value requested.	
Current Sync State	string	The textual equivalent (Value) for the current StorageSynchronized.SyncState value	

Table 77 describes the error properties.

Table 77 - Error Properties for Invalid State Transition

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Reference to the StorageSynchronized instance in question.	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.25 Message: Invalid SAP for Method

Owning Entity: SNIA

Message ID: DRM25

Message Format String: Invalid type of copy services host. The host must be a <Host Type>

A message indicating an invalid copy services host. Table 78 describes the message arguments.

Table 78 - Invalid SAP for Method Message Arguments

Message Argument	Data Type	Description	Possible Values
Host Type	string	The type of copy services on which the method was invoked.	source
			target

Table 79 describes the error properties.

Table 79 - Error Properties for Invalid SAP for Method

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Reference to the Computer System host which is of the wrong type.	Existence is required
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.26 Message: Resource Not Available

Owning Entity: SNIA

Message ID: DRM26

Message Format String: <Resource Needed>

A message indicating that a resource was not available for replication. Table 80 describes the message arguments.

Table 80 - Resource Not Available Message Arguments

Message Argument	Data Type	Description	Possible Values
Resource Needed	string		No replication log available.
			Special replica pool required.

Table 81 describes the error properties.

Table 81 - Error Properties for Resource Not Available

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Nothing to reference.	Existence is discouraged
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.2.27 Message: Resource Limit Exceeded

Owning Entity: SNIA

Message ID: DRM27

Message Format String: <Reason>

A message indicating a resource limit has been reached. Table 82 describes the message arguments.

Table 82 - Resource Limit Exceeded Message Arguments

Message Argument	Data Type	Description	Possible Values
Reason	string	The reasons for the lack of resources for copy services operation.	Insufficient pool space.
			Maximum replication depth exceeded.
			Maxium replicas exceeded for source element.

Table 83 describes the error properties.

Table 83 - Error Properties for Resource Limit Exceeded

Property	Value	Description
CIMSTATUSCODE	1(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	4(Software Error)	Existence is required
ERROR_SOURCE	Nothing to reference.	Existence is discouraged
PERCEIVED_SEVERITY	2(Information)	Existence is required

EXPERIMENTAL

8.3.2.28 Message: Thin Provision Capacity Warning

Owning Entity: SNIA

Message ID: DRM28

Message Format String: <Element type> with identifier <Device or Pool ID or Name> is nearing its available capacity limit.

The actual capacity of a volume or pool is nearing a limit (e.g., actual usage of containing pool is nearing SpaceLimit). Table 84 describes the message arguments.

Table 84 - Thin Provision Capacity Warning Message Arguments

Message Argument	Data Type	Description	Possible Values
Element type	string	A 'volume', 'pool' or 'filesystem'	volume
			pool
			filesystem
Device or Pool ID or Name	string	Disk Name.	

Table 85 describes the alerts that are associated with this message.

Table 85 - Thin Provision Capacity Warning Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The pool, volume or filesystem ID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

8.3.2.29 Message: Provision Capacity Critical

Owning Entity: SNIA

Message ID: DRM29

Message Format String: <Element type> with identifier <Device or Pool ID or Name> has exceeded the available capacity limit.

the actual capacity of a volume or pool has reached a limit (e.g., actual usage of containing pool is equal to SpaceLimit). Write commands from hosts to the volume or pool are failing. . Table 86 describes the message arguments.

Table 86 - Provision Capacity Critical Message Arguments

Message Argument	Data Type	Description	Possible Values
Element type	string	A 'volume', 'pool' or 'filesystem'	volume
			pool
			filesystem
Device or Pool ID or Name	string	Disk Name.	

Message: Thin Provision Capacity Okay

Table 87 describes the alerts that are associated with this message.

Table 87 - Provision Capacity Critical Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The pool, volume or filesystem ID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.2.30 Message: Thin Provision Capacity Okay

Owning Entity: SNIA

Message ID: DRM30

Message Format String: <Thin element type> with identifier <Device or Pool ID> capacity condition cleared.

The actual capacity of a volume or pool is no longer in a capacity warning or critical state. Table 88 describes the message arguments.

Table 88 - Thin Provision Capacity Okay Message Arguments

Message Argument	Data Type	Description	Possible Values
Thin element type	string	A 'volume', 'pool' or 'filesystem'	volume
			pool
			filesystem
Device or Pool ID	string	Disk Name.	

Table 89 describes the alerts that are associated with this message.

Table 89 - Thin Provision Capacity Okay Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The pool, volume or filesystem ID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.31 Message: Masking Group Membership Changed

Owning Entity: SNIA

Message ID: DRM31

Message Format String: There is a change in membership of masking group with identifier <InstanceID> , and with ElementName <ElementName>

The membership of a masking group has changed. Table 90 describes the message arguments.

Table 90 - Masking Group Membership Changed Message Arguments

Message Argument	Data Type	Description	Possible Values
InstanceID	string	The instance ID of masking group	InstanceID
ElementName	string	The ElementName of masking group	ElementName

Table 91 describes the alerts that are associated with this message.

Table 91 - Masking Group Membership Changed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The masking group object name.
ALERT_TYPE	Y	2	Model Change
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.32 Message: StorageVolume Relocation Starts

Owning Entity: SNIA

Message ID: DRM32

Message Format String: Relocation is starting for the Storage Volume with identifier <StorageVolume DeviceID>

The relocation of a storage volume is starting. Table 92 describes the message arguments.

Table 92 - StorageVolume Relocation Starts Message Arguments

Message Argument	Data Type	Description	Possible Values
StorageVolume DeviceID	string	DeviceID.	

Table 93 describes the alerts that are associated with this message.

Table 93 - StorageVolume Relocation Starts Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The StorageVolume DeviceID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.33 Message: StorageVolume Relocation Ends

Owning Entity: SNIA

Message ID: DRM33

Message Format String: Relocation has ended for the Storage Volume with identifier <StorageVolume DeviceID>

Message: StoragePool Relocation Starts

The relocation of a storage volume has ended. Table 94 describes the message arguments.

Table 94 - StorageVolume Relocation Ends Message Arguments

Message Argument	Data Type	Description	Possible Values
StorageVolume DeviceID	string	DeviceID.	

Table 95 describes the alerts that are associated with this message.

Table 95 - StorageVolume Relocation Ends Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The StorageVolume DeviceID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.34 Message: StoragePool Relocation Starts

Owning Entity: SNIA

Message ID: DRM34

Message Format String: Relocation is starting for the Storage Pool with identifier <StoragePool PoolID>

The relocation of a storage pool is starting. Table 96 describes the message arguments.

Table 96 - StoragePool Relocation Starts Message Arguments

Message Argument	Data Type	Description	Possible Values
StoragePool PoolID	string	PoolID.	

Table 97 describes the alerts that are associated with this message.

Table 97 - StoragePool Relocation Starts Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The StoragePool PoolID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.35 Message: StoragePool Relocation Ends

Owning Entity: SNIA

Message ID: DRM35

Message Format String: Relocation has ended for the Storage Pool with identifier <StoragePool PoolID>

The relocation of a storage pool has ended. Table 98 describes the message arguments.

Table 98 - StoragePool Relocation Ends Message Arguments

Message Argument	Data Type	Description	Possible Values
StoragePool PoolID	string	PoolID.	

Table 99 describes the alerts that are associated with this message.

Table 99 - StoragePool Relocation Ends Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The StoragePool PoolID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.36 Message: LogicalDisk Relocation Starts

Owning Entity: SNIA

Message ID: DRM36

Message Format String: Relocation is starting for the Logical Disk with identifier <LogicalDisk DeviceID>

The relocation of a logical disk is starting. Table 100 describes the message arguments.

Table 100 - LogicalDisk Relocation Starts Message Arguments

Message Argument	Data Type	Description	Possible Values
LogicalDisk DeviceID	string	DeviceID.	

Table 101 describes the alerts that are associated with this message.

Table 101 - LogicalDisk Relocation Starts Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The LogicalDisk DeviceID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.2.37 Message: LogicalDisk Relocation Ends

Owning Entity: SNIA

Message ID: DRM37

Message Format String: Relocation has ended for the Logical Disk with identifier <LogicalDisk DeviceID>

Message: Volume or pool degraded

The relocation of a logical disk has ended. Table 102 describes the message arguments.

Table 102 - LogicalDisk Relocation Ends Message Arguments

Message Argument	Data Type	Description	Possible Values
LogicalDisk DeviceID	string	DeviceID.	

Table 103 describes the alerts that are associated with this message.

Table 103 - LogicalDisk Relocation Ends Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The LogicalDisk DeviceID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

8.3.2.38 Message: Volume or pool degraded

Owning Entity: SNIA

Message ID: DRM38

Message Format String: Volume or pool <Element type> with identifier <Device or Pool ID> is in a degraded state.

The status of a volume or pool is degraded. . Table 104 describes the message arguments.

Table 104 - Volume or pool degraded Message Arguments

Message Argument	Data Type	Description	Possible Values
Element type	string	A 'volume' or 'pool'	volume
			pool
Device or Pool ID	string	Disk Name.	

Table 105 describes the alerts that are associated with this message.

Table 105 - Volume or pool degraded Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The pool or volume ID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.2.39 Message: Volume or pool failed

Owning Entity: SNIA

Message ID: DRM39

Message Format String: Volume or pool <Element type> with identifier <Device or Pool ID> has failed and is not accessible.

The status of a volume or pool is failed. . Table 106 describes the message arguments.

Table 106 - Volume or pool failed Message Arguments

Message Argument	Data Type	Description	Possible Values
Element type	string	A 'volume' or 'pool'	volume
			pool
Device or Pool ID	string	Disk Name.	

Table 107 describes the alerts that are associated with this message.

Table 107 - Volume or pool failed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The pool or volume ID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.2.40 Message: Volume or pool issues cleared

Owning Entity: SNIA

Message ID: DRM40

Message Format String: Volume or pool <Element type> with identifier <Device or Pool ID> is no longer degraded.

The status of a volume or pool is normal. . Table 108 describes the message arguments.

Table 108 - Volume or pool issues cleared Message Arguments

Message Argument	Data Type	Description	Possible Values
Element type	string	A 'volume' or 'pool'	volume
			pool
Device or Pool ID	string	Disk Name.	

Message: The StoragePool is healthy

Table 109 describes the alerts that are associated with this message.

Table 109 - Volume or pool issues cleared Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The pool or volume ID.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.41 Message: The StoragePool is healthy

Owning Entity: SNIA

Message ID: DRM101

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has found that the pool has an OK OperationalStatus.

The test ran to completion and found the OperationalStatus to be OK. Table 110 describes the message arguments.

Table 110 - The StoragePool is healthy Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test The ElementName of the pool under test A unique user friendly name not in the model (such as, asset name)

Table 111 describes the alerts that are associated with this message.

Table 111 - The StoragePool is healthy Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (StoragePool is OK)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.42 Message: StoragePool is dependent on an element with problems

Owning Entity: SNIA

Message ID: DRM102

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of <OperationalStatus1> , because <Named Element> <Element Moniker> has an OperationalStatus of <OperationalStatus2>

The test found that an element the pool is dependent on has a non-OK OperationalStatus. Table 112 describes the message arguments.

Table 112 - StoragePool is dependent on an element with problems Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
OperationalStatus1	string	The OperationalStatus of the pool under test.	
Named Element	string	The common name (e.g. Storage Pool or Disk Drive) for the non-OK element	
Element Moniker	string	A unique name for the contributing element with the non-OK state.	The object path of the contributing element
			The ElementName of the contributing element
			A unique user friendly name not in the model (such as, asset name)
OperationalStatus2	string	The OperationalStatus of the contributing element.	

Table 113 describes the alerts that are associated with this message.

Table 113 - StoragePool is dependent on an element with problems Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (StoragePool element problem)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.43 Message: The StoragePool is being serviced

Owning Entity: SNIA

Message ID: DRM103

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of <OperatonaStatus> , because it is being <Service Action>

The test found that the StoragePool is being serviced, which results in the pool OperationalStatus. Table 114 describes the message arguments.

Table 114 - The StoragePool is being serviced Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
OperationalStatus	string	The OperationalStatus of the storage pool under test.	
Service Action	string	The temporary service that is in progress.	Rebuilt
			Relocated
			Tested
			Serviced

Table 115 describes the alerts that are associated with this message.

Table 115 - The StoragePool is being serviced Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (StoragePool servicing in progress)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.44 Message: The OperationalStatus of the Pool is impacting an element allocated from it

Owning Entity: SNIA

Message: The OperationalStatus of the Pool is impacting an element allocated from it

Message ID: DRM104

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of <Operatona1Status1> that is causing <Named Element> <Element Moniker> to have an OperationalStatus of <Operatona1Status2>

The test found that the OperationalStatus of the StoragePool is impacting an element that is allocated from the pool. Table 116 describes the message arguments.

Table 116 - The OperationalStatus of the Pool is impacting an element allocated from it Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
Operatona1Status1	string	The OperationalStatus of the storage pool under test.	
Named Element	string	The common name (e.g. Storage Pool or Storage Volume) for the non-OK element	Storage Pool
			Storage Volume
			Logical Disk
			Filesystem
Element Moniker	string	A unique name for the allocated element with the non-OK state.	The object path of the allocated element
			The ElementName of the allocated element
			A unique user friendly name not in the model (such as, asset name)
Operatona1Status2	string	The OperationalStatus of the element allocated from the pool under test.	

Table 117 describes the alerts that are associated with this message.

Table 117 - The OperationalStatus of the Pool is impacting an element allocated from it Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Element Impacted)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

Message: The StoragePool OperationalStatus may be corrected by applying a spare

EXPERIMENTAL

8.3.2.45 Message: The StoragePool OperationalStatus may be corrected by applying a spare

Owning Entity: SNIA

Message ID: DRM105

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of <OperationalStatus> can be fixed by applying one of the following spare <Named Element> <List of Spares>

The test found that applying a spare will correct the OperationalStatus of the Pool. Table 118 describes the message arguments.

Table 118 - The StoragePool OperationalStatus may be corrected by applying a spare Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
OperationalStatus	string	The OperationalStatus of the storage pool under test.	
Named Element	string	The common name (e.g. Storage Extent or Disk Drive) for the spare element	Storage Extent
			Disk Drive
List of Spares	string	An array of element monikers of the available spares.	The object path of the spare element
			The ElementName of the spare element
			A unique user friendly name not in the model (such as, asset name)

Table 119 describes the alerts that are associated with this message.

Table 119 - The StoragePool OperationalStatus may be corrected by applying a spare Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Deploy Spare)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

Message: The StoragePool OperationalStatus may be corrected by relocating the pool

EXPERIMENTAL

8.3.2.46 Message: The StoragePool OperationalStatus may be corrected by relocating the pool

Owning Entity: SNIA

Message ID: DRM106

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of <OperatonaStatus> can be fixed by relocating the storage pool.

The test found that relocating the pool will correct the OperationalStatus of the pool. Table 120 describes the message arguments.

Table 120 - The StoragePool OperationalStatus may be corrected by relocating the pool Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
OperatonaStatus	string	The OperationalStatus of the storage pool under test.	

Table 121 describes the alerts that are associated with this message.

Table 121 - The StoragePool OperationalStatus may be corrected by relocating the pool Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (StoragePool may be relocated)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.47 Message: Pool experiencing interference from system workload

Owning Entity: SNIA

Message ID: DRM107

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of <OperatonaStatus> is experiencing interference from other system workloads.

Message: Pool performance degraded by component element

The test found that the storage pool has its operational status because of interference from system workloads. Table 122 describes the message arguments.

Table 122 - Pool experiencing interference from system workload Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
OperationalStatus	string	The OperationalStatus of the storage pool under test.	

Table 123 describes the alerts that are associated with this message.

Table 123 - Pool experiencing interference from system workload Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Workload Interference)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.48 Message: Pool performance degraded by component element

Owning Entity: SNIA

Message ID: DRM108

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> has an OperationalStatus of degraded due to conditions in <Named Element> <Element Moniker> has an OperationalStatus of <OperationalStatus>

Message: Pool degraded due to loss of RAID protection

The test found that the activity in the storage pool may be experiencing performance problems because a component element (e.g., parent storage pool or disk drive) has a non-OK OperationalStatus. Table 124 describes the message arguments.

Table 124 - Pool performance degraded by component element Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)
Named Element	string	The common name (e.g. Storage Extent or Disk Drive) for the non-OK element	Storage Extent
			Disk Drive
			Back-end port
			Storage Pool
Element Moniker	string	A unique name for the contributing element with the non-OK state.	The object path of the contributing element
			The ElementName of the contributing element
			A unique user friendly name not in the model (such as, asset name)
OperationalStatus	string	The OperationalStatus of the contributing element.	

Table 125 describes the alerts that are associated with this message.

Table 125 - Pool performance degraded by component element Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Degraded by Element)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.49 Message: Pool degraded due to loss of RAID protection

Owning Entity: SNIA

Message ID: DRM109

Message: Pool degraded due to loss of port redundancy

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> is degraded due to loss of RAID protection.

The test found that the storage pool is degraded due to the loss of RAID protection (PackageRedundancy or DataRedundancy). Table 126 describes the message arguments.

Table 126 - Pool degraded due to loss of RAID protection Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)

Table 127 describes the alerts that are associated with this message.

Table 127 - Pool degraded due to loss of RAID protection Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Loss of RAID protection)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.50 Message: Pool degraded due to loss of port redundancy

Owning Entity: SNIA

Message ID: DRM110

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> is degraded due to loss of port redundancy for disk access.

The test found that the storage pool is degraded due to disk access degradation due to a failing port. Table 128 describes the message arguments.

Table 128 - Pool degraded due to loss of port redundancy Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)

Table 129 describes the alerts that are associated with this message.

Table 129 - Pool degraded due to loss of port redundancy Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Loss of Port redundancy)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.2.51 Message: Pool predicting failure due lack of available capacity

Owning Entity: SNIA

Message ID: DRM111

Message Format String: The <Test> test on the selected storage pool to test <Pool Moniker> is predicting failure because the available capacity is low.

The test found that the storage pool is predicting failure because it is running low on available capacity. Table 130 describes the message arguments.

Table 130 - Pool predicting failure due lack of available capacity Message Arguments

Message Argument	Data Type	Description	Possible Values
Test	string	The Name property value of the StoragePoolDiagnosticTest instance invoked.	
Pool Moniker	string	A unique name for the pool under test that was specified.	The object path of the pool under test

Table 130 - Pool predicting failure due lack of available capacity Message Arguments

Message Argument	Data Type	Description	Possible Values
			The ElementName of the pool under test
			A unique user friendly name not in the model (such as, asset name)

Table 131 describes the alerts that are associated with this message.

Table 131 - Pool predicting failure due lack of available capacity Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The object name of the pool under test.
ALERT_TYPE	Y	1	Other (Low Available Capacity)
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

8.3.3 Fabric Messages

8.3.3.1 Message: Zone Database Changed

Owning Entity: SNIA

Message ID: FC1

Message Format String: Zone database changed for <Fabric Identity Type> <WWN>

An Indication when the fabric or switch has determined that the Zone Database has been modified. Table 132 describes the message arguments.

Table 132 - Zone Database Changed Message Arguments

Message Argument	Data Type	Description	Possible Values
Fabric Identity Type	string	Defines the type of fabric entity names by the following WWN.	fabric
			switch
WWN	string	World Wide name identifier. The required form of the WWN is defined by this regular expression, <code>"^[0123456789ABCDEF]{16}\$"</code>	

Table 133 describes the alerts that are associated with this message.

Table 133 - Zone Database Changed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		A reference to the switch or fabric which is named by the WWN.

Table 133 - Zone Database Changed Alert Information

Name	Req	Value	Description
ALERT_TYPE	Y		Environmental Alert
PERCEIVED_SEVERITY	Y		Informational

8.3.3.2 Message: ZoneSet Activated

Owning Entity: SNIA

Message ID: FC2

Message Format String: ZoneSet <ZoneSet Name> was activated for fabric <WWN>

An Indication when the fabric has determined that a ZoneSet has been activated. Table 134 describes the message arguments.

Table 134 - ZoneSet Activated Message Arguments

Message Argument	Data Type	Description	Possible Values
ZoneSet Name	string	CIM_ZoneSet.ElementName attribute	
WWN	string	World Wide name identifier. The required form of the WWN is defined by this regular expression, "^[0123456789ABCDEF]{16}\$"	

Table 135 describes the alerts that are associated with this message.

Table 135 - ZoneSet Activated Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		A reference to the fabric which is named by the WWN.
ALERT_TYPE	Y		Environmental Error
PERCEIVED_SEVERITY	Y		Minor

EXPERIMENTAL

8.3.3.3 Message: Session Locked

Owning Entity: SNIA

Message ID: FC3

Message Format String: Operation blocked by session lock.

Message: Session Aborted

Table 136 describes the error properties.

Table 136 - Error Properties for Session Locked

Property	Value	Description
CIMSTATUSCODE	(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	(Software Error)	Existence is required
ERROR_SOURCE		Existence is required
PERCEIVED_SEVERITY	(Degraded/Warning)	Existence is required

EXPERIMENTAL

EXPERIMENTAL

8.3.3.4 Message: Session Aborted

Owning Entity: SNIA

Message ID: FC4

Message Format String: Operation by another client caused the session to be aborted.

Table 137 describes the error properties.

Table 137 - Error Properties for Session Aborted

Property	Value	Description
CIMSTATUSCODE	(CIM_ERR_FAILED)	Existence is required
ERROR_TYPE	(Software Error)	Existence is required
ERROR_SOURCE		Existence is required
PERCEIVED_SEVERITY	(Degraded/Warning)	Existence is required

EXPERIMENTAL

8.3.3.5 Message: Switch Status Changed

Owning Entity: SNIA

Message ID: FC5

Message Format String: Switch <Switch Unique Identifier> in Fabric <Fabric Name> status changed to <Switch OperationalStatus>

The fabric has detected a change in status of a switch in the fabric. Table 138 describes the message arguments.

Table 138 - Switch Status Changed Message Arguments

Message Argument	Data Type	Description	Possible Values
Switch Unique Identifier	string	The Switch Name (WWN).	
Fabric Name	string	Fabric name. Typically the principal switch's WWN	
Switch OperationalStatus	string	Switch Status	

Table 139 describes the alerts that are associated with this message.

Table 139 - Switch Status Changed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The Fabric
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y		Major

8.3.3.6 Message: Fabric Merge/Segmentation

Owning Entity: SNIA

Message ID: FC6

Message Format String: <Fabric Name> has detected a <Fabric Change>

The fabric has detected either two fabrics have merged into a single fabric or a single fabric has segmented. . Table 140 describes the message arguments.

Table 140 - Fabric Merge/Segmentation Message Arguments

Message Argument	Data Type	Description	Possible Values
Fabric Name	string	Fabric name. Typically the principal switch's WWN	
Fabric Change	string	A value of merge or segmentation	merge segmentation

Table 141 describes the alerts that are associated with this message.

Table 141 - Fabric Merge/Segmentation Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The SAN
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y		Minor

8.3.3.7 Message: Switch Added/Removed

Owning Entity: SNIA

Message: Fabric Added/Removed

Message ID: FC7

Message Format String: The fabric <Fabric Name> has detected switch <Switch Unique Identifier> has been <Fabric Change Type>

A Switch has been added or removed from the fabric. Table 142 describes the message arguments.

Table 142 - Switch Added/Removed Message Arguments

Message Argument	Data Type	Description	Possible Values
Fabric Name	string	Fabric name. Typically the principal switch's WWN	
Switch Unique Identifier	string	The Switch Name (WWN).	
Fabric Change Type	string	A value of added or removed	added removed

Table 143 describes the alerts that are associated with this message.

Table 143 - Switch Added/Removed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The Fabric
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y		Minor

8.3.3.8 Message: Fabric Added/Removed

Owning Entity: SNIA

Message ID: FC8

Message Format String: Fabric <Fabric Identifier> was <Change Type>

The agent has detected the addition or removal of a fabric from the SAN. This message can also be used for Virtual Fabrics. Table 144 describes the message arguments.

Table 144 - Fabric Added/Removed Message Arguments

Message Argument	Data Type	Description	Possible Values
Fabric Identifier	string	The Fabric Identity	
Change Type	string	A value of Added or Removed	added removed

Table 145 describes the alerts that are associated with this message.

Table 145 - Fabric Added/Removed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The SAN

Table 145 - Fabric Added/Removed Alert Information

Name	Req	Value	Description
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y		Major

EXPERIMENTAL

8.3.3.9 Message: Security Policy change

Owning Entity: SNIA

Message ID: FC9

Message Format String: Fabric Security Policy changed in <Fabric Name>

The fabric has detected a change in the Security Database. Table 146 describes the message arguments.

Table 146 - Security Policy change Message Arguments

Message Argument	Data Type	Description	Possible Values
Fabric Name	string	Fabric name. Typically the principal switch's WWN	

Table 147 describes the alerts that are associated with this message.

Table 147 - Security Policy change Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The Fabric
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y		Minor

EXPERIMENTAL

8.3.4 Filesystem Messages

EXPERIMENTAL

8.3.4.1 Message: System OperationalStatus Bellwether

Owning Entity: SNIA

Message ID: FSM1

Message Format String: The OperationalStatus of the <System Name> system has changed. Related elements will not report the change in their OperationalStatus.

A message indicating the change in OperationalStatus of a system is a bellwether alert. Table 148 describes the message arguments.

Table 148 - System OperationalStatus Bellwether Message Arguments

Message Argument	Data Type	Description	Possible Values
System Name	string	The Name property of the system whose OperationalStatus has changed.	The Name of a NAS System
			The Name of a Component System
			The Name of a Base Server System
			The Name of a Virtual File Server System

Table 149 describes the alerts that are associated with this message.

Table 149 - System OperationalStatus Bellwether Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.2 Message: NetworkPort OperationalStatus Bellwether

Owning Entity: SNIA

Message ID: FSM2

Message Format String: The OperationalStatus of the <NetworkPort Name> network port on the <System Name> system has changed. Related elements will not report the change in their OperationalStatus.

A message indicating the change in OperationalStatus of a NetworkPort is a bellwether alert. Table 150 describes the message arguments.

Table 150 - NetworkPort OperationalStatus Bellwether Message Arguments

Message Argument	Data Type	Description	Possible Values
NetworkPort Name	string	The ElementName property of the network port whose OperationalStatus has changed.	
System Name	string	The Name property of the system on which the port exists	The Name of a NAS System
			The Name of a Component System
			The Name of a Base Server System
			The Name of a Virtual File Server System

Table 151 describes the alerts that are associated with this message.

Table 151 - NetworkPort OperationalStatus Bellwether Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.3 Message: LogicalDisk OperationalStatus Bellwether

Owning Entity: SNIA

Message ID: FSM3

Message Format String: The OperationalStatus of the <LogicalDisk Name> logical disk on the <System Name> system has changed. Related elements will not report the change in their OperationalStatus.

A message indicating the change in OperationalStatus of a LogicalDisk is a bellwether alert. Table 152 describes the message arguments.

Table 152 - LogicalDisk OperationalStatus Bellwether Message Arguments

Message Argument	Data Type	Description	Possible Values
LogicalDisk Name	string	The Name property of the logical disk whose OperationalStatus has changed.	
System Name	string	The Name property of the system on which the logical disk is known.	The Name of a NAS System
			The Name of a Component System
			The Name of a Base Server System
			The Name of a Virtual File Server System

Table 153 describes the alerts that are associated with this message.

Table 153 - LogicalDisk OperationalStatus Bellwether Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.4 Message: CopyState is set to Broken

Owning Entity: SNIA

Message ID: FSM4

Message Format String: CopyState of an alerting Managed Element, which is an instance of CIM_Synchronized containing a SystemElement.ElementName= <Source Element Name> and SyncedElement.ElementName= <Target Element Name> , is set to Broken.

CopyState is set to Broken. Table 154 describes the message arguments.

Table 154 - CopyState is set to Broken Message Arguments

Message Argument	Data Type	Description	Possible Values
Source Element Name	string	The textual equivalent (Value) for SystemElement.ElementName value	
Target Element Name	string	The textual equivalent (Value) for SyncedElement.ElementName value	

Table 155 describes the alerts that are associated with this message.

Table 155 - CopyState is set to Broken Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.5 Message: Not Enough Space

Owning Entity: SNIA

Message ID: FSM5

Message Format String: Remaining pool space either below the warning threshold set for the <StoragePool Name> or there is no remaining space in the <StoragePool Name>

Not Enough Space. Table 156 describes the message arguments.

Table 156 - Not Enough Space Message Arguments

Message Argument	Data Type	Description	Possible Values
StoragePool Name	string	The textual equivalent (Value) for StoragePool.Name	
StoragePool Name	string	The textual equivalent (Value) for StoragePool.Name	

Table 157 describes the alerts that are associated with this message.

Table 157 - Not Enough Space Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.6 Message: The changes in RemoteReplicationCollection

Owning Entity: SNIA

Message ID: FSM6

Message Format String: The collection of the paths that provide the access to a remote system for replication operations in <RemoteReplicationCollection Name> are changed

The changes in RemoteReplicationCollection. Table 158 describes the message arguments.

Table 158 - The changes in RemoteReplicationCollection Message Arguments

Message Argument	Data Type	Description	Possible Values
RemoteReplicationCollecti on Name	string	The textual equivalent (Value) for Synchronized.SystemElement value	

Message: The changes in ProtocolEndpoint

Table 159 describes the alerts that are associated with this message.

Table 159 - The changes in RemoteReplicationCollection Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.7 Message: The changes in ProtocolEndpoint

Owning Entity: SNIA

Message ID: FSM7

Message Format String: The networking protocol <ProtocolEndpoint Name> which enables a replication service to reach a remote element is changed.

The changes in ProtocolEndpoint. Table 160 describes the message arguments.

Table 160 - The changes in ProtocolEndpoint Message Arguments

Message Argument	Data Type	Description	Possible Values
ProtocolEndpoint Name	string	The textual equivalent (Value) for ProtocolEndpoint.Name	

Table 161 describes the alerts that are associated with this message.

Table 161 - The changes in ProtocolEndpoint Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.8 Message: CopyState is set to Broken

Owning Entity: SNIA

Message ID: FSM8

Message: CopyState is set to Invalid

Message Format String: CopyState of an alerting Managed Element, which is an instance of CIM_Synchronized containing a SystemElement.ElementName= <Source Element Name> and SyncedElement.ElementName= <Target Element Name> , is set to Fractured.

CopyState is set to Fractured. Table 162 describes the message arguments.

Table 162 - CopyState is set to Broken Message Arguments

Message Argument	Data Type	Description	Possible Values
Source Element Name	string	The textual equivalent (Value) for SystemElement.ElementName value	
Target Element Name	string	The textual equivalent (Value) for SyncedElement.ElementName value	

Table 163 describes the alerts that are associated with this message.

Table 163 - CopyState is set to Broken Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.9 Message: CopyState is set to Invalid

Owning Entity: SNIA

Message ID: FSM9

Message Format String: CopyState of an alerting Managed Element, which is an instance of CIM_Synchronized containing a SystemElement.ElementName= <Source Element Name> and SyncedElement.ElementName= <Target Element Name> , is set to Invalid.

CopyState is set to Invalid. Table 164 describes the message arguments.

Table 164 - CopyState is set to Invalid Message Arguments

Message Argument	Data Type	Description	Possible Values
Source Element Name	string	The textual equivalent (Value) for SystemElement.ElementName value	
Target Element Name	string	The textual equivalent (Value) for SyncedElement.ElementName value	

Message: CopyState is set to Inactive

Table 165 describes the alerts that are associated with this message.

Table 165 - CopyState is set to Invalid Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.10 Message: CopyState is set to Inactive

Owning Entity: SNIA

Message ID: FSM10

Message Format String: CopyState of an alerting Managed Element, which is an instance of CIM_Synchronized containing a SystemElement.ElementName= <Source Element Name> and SyncedElement.ElementName= <Target Element Name> , is set to Inactive.

CopyState is set to Inactive. Table 166 describes the message arguments.

Table 166 - CopyState is set to Inactive Message Arguments

Message Argument	Data Type	Description	Possible Values
Source Element Name	string	The textual equivalent (Value) for SystemElement.ElementName value	
Target Element Name	string	The textual equivalent (Value) for SyncedElement.ElementName value	

Table 167 describes the alerts that are associated with this message.

Table 167 - CopyState is set to Inactive Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Minor

EXPERIMENTAL

EXPERIMENTAL

8.3.4.11 Message: CopyState is set to Split

Owning Entity: SNIA

Message: CopyState alert has been cleared

Message ID: FSM11

Message Format String: CopyState of an alerting Managed Element, which is an instance of CIM_Synchronized containing a SystemElement.ElementName= <Source Element Name> and SyncedElement.ElementName= <Target Element Name> , is set to Split.

CopyState is set to Split. Table 168 describes the message arguments.

Table 168 - CopyState is set to Split Message Arguments

Message Argument	Data Type	Description	Possible Values
Source Element Name	string	The textual equivalent (Value) for SystemElement.ElementName value	
Target Element Name	string	The textual equivalent (Value) for SyncedElement.ElementName value	

Table 169 describes the alerts that are associated with this message.

Table 169 - CopyState is set to Split Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Minor

EXPERIMENTAL

EXPERIMENTAL

8.3.4.12 Message: CopyState alert has been cleared

Owning Entity: SNIA

Message ID: FSM12

Message Format String: CopyState of an alerting Managed Element, which is an instance of CIM_Synchronized containing a SystemElement.ElementName= <Source Element Name> and SyncedElement.ElementName= <Target Element Name> , is set to a non-adverse condition.

CopyState adverse condition cleared. Table 170 describes the message arguments.

Table 170 - CopyState alert has been cleared Message Arguments

Message Argument	Data Type	Description	Possible Values
Source Element Name	string	The textual equivalent (Value) for SystemElement.ElementName value	
Target Element Name	string	The textual equivalent (Value) for SyncedElement.ElementName value	

Message: Available Space Changed

Table 171 describes the alerts that are associated with this message.

Table 171 - CopyState alert has been cleared Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

EXPERIMENTAL

EXPERIMENTAL

8.3.4.13 Message: Available Space Changed

Owning Entity: SNIA

Message ID: FSM13

Message Format String: The available space of the filesystem <Filesystem Name> has changed

The available space of a filesystem has changed. Table 172 describes the message arguments.

Table 172 - Available Space Changed Message Arguments

Message Argument	Data Type	Description	Possible Values
Filesystem Name	string	The textual equivalent (Value) for Filesystem.Name	

Table 173 describes the alerts that are associated with this message.

Table 173 - Available Space Changed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.4.14 Message: Filesystem Inaccessible

Owning Entity: SNIA

Message ID: FSM14

Message Format String: The filesystem <Filesystem Name> is not accessible

The filesystem is not accessible. Table 174 describes the message arguments.

Table 174 - Filesystem Inaccessible Message Arguments

Message Argument	Data Type	Description	Possible Values
Filesystem Name	string	The textual equivalent (Value) for Filesystem.Name	

Table 175 describes the alerts that are associated with this message.

Table 175 - Filesystem Inaccessible Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.4.15 Message: Filesystem is Online

Owning Entity: SNIA

Message ID: FSM15

Message Format String: The filesystem <Filesystem Name> is back online

The filesystem is online. Table 176 describes the message arguments.

Table 176 - Filesystem is Online Message Arguments

Message Argument	Data Type	Description	Possible Values
Filesystem Name	string	The textual equivalent (Value) for Filesystem.Name	

Table 177 describes the alerts that are associated with this message.

Table 177 - Filesystem is Online Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

EXPERIMENTAL

8.3.4.16 Message: Fileshare is degraded

Owning Entity: SNIA

Message ID: FSM16

Message Format String: The fileshare <Fileshare Name> is in a degraded state

The fileshare state is degraded. Table 178 describes the message arguments.

Table 178 - Fileshare is degraded Message Arguments

Message Argument	Data Type	Description	Possible Values
Fileshare Name	string	The textual equivalent (Value) for Fileshare.Name	

Table 179 describes the alerts that are associated with this message.

Table 179 - Fileshare is degraded Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.4.17 Message: Fileshare in normal state

Owning Entity: SNIA

Message ID: FSM17

Message Format String: The fileshare <Fileshare Name> has returned to normal conditions.

The fileshare has returned to normal operations. Table 180 describes the message arguments.

Table 180 - Fileshare in normal state Message Arguments

Message Argument	Data Type	Description	Possible Values
Fileshare Name	string	The textual equivalent (Value) for Fileshare.Name	

Table 181 describes the alerts that are associated with this message.

Table 181 - Fileshare in normal state Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

8.3.5 Host Messages

8.3.5.1 Message: Required Firmware Version

Owning Entity: SNIA

Message ID: Host1

Message Format String: Controller firmware is older than required. Current Version: <Current Version>

Minimum required version: <Minimum required version>

A message indicating the controller firmware is older than required. Table 182 describes the message arguments.

Table 182 - Required Firmware Version Message Arguments

Message Argument	Data Type	Description	Possible Values
Current Version	string	The current firmware version number.	
Minimum required version	string	The minimum required version number	

Table 183 describes the alerts that are associated with this message.

Table 183 - Required Firmware Version Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The controller.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

8.3.5.2 Message: Recommended Firmware Version

Owning Entity: SNIA

Message ID: Host2

Message Format String: Controller firmware is older than recommended. Current Version: <Current Version>

Minimum recommended version: <Minimum recommended version>

Message: Controller OK

A message indicating the controller firmware is older than recommended. Table 184 describes the message arguments.

Table 184 - Recommended Firmware Version Message Arguments

Message Argument	Data Type	Description	Possible Values
Current Version	string	The current firmware version number.	
Minimum recommended version	string	The minimum recommended version number	

Table 185 describes the alerts that are associated with this message.

Table 185 - Recommended Firmware Version Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The controller.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Minor

8.3.5.3 Message: Controller OK

Owning Entity: SNIA

Message ID: Host3

Message Format String: Controller health is ok. Controller Name: <Controller Name>

Table 186 describes the message arguments.

Table 186 - Controller OK Message Arguments

Message Argument	Data Type	Description	Possible Values
Controller Name	string	Controller Name.	

Table 187 describes the alerts that are associated with this message.

Table 187 - Controller OK Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The controller.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

8.3.5.4 Message: Controller not OK

Owning Entity: SNIA

Message ID: Host4

Message Format String: Controller health is not ok. Controller Name: <Controller Name>

Table 188 describes the message arguments.

Table 188 - Controller not OK Message Arguments

Message Argument	Data Type	Description	Possible Values
Controller Name	string	Controller Name.	

Table 189 describes the alerts that are associated with this message.

Table 189 - Controller not OK Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The controller.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

8.3.5.5 Message: Bus rescan complete

Owning Entity: SNIA

Message ID: Host5

Message Format String: Bus rescan complete

Table 190 describes the alerts that are associated with this message.

Table 190 - Bus rescan complete Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The controller.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Minor

8.3.5.6 Message: Disk initialize Failed

Owning Entity: SNIA

Message ID: Host6

Message Format String: Disk Initialize Failed. Disk name: <Disk Name>

Table 191 describes the message arguments.

Table 191 - Disk initialize Failed Message Arguments

Message Argument	Data Type	Description	Possible Values
Disk Name	string	Disk Name.	

Message: Read Warning

Table 192 describes the alerts that are associated with this message.

Table 192 - Disk initialize Failed Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		The system containing the controller.
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Major

8.3.6 Media Library Messages

8.3.6.1 Message: Read Warning

Owning Entity: SNIA

Message ID: SML1

Message Format String: The drive is having severe trouble reading.

Table 193 describes the alerts that are associated with this message.

Table 193 - Read Warning Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.2 Message: Write Warning

Owning Entity: SNIA

Message ID: SML2

Message Format String: The drive is having severe trouble writing.

Table 194 describes the alerts that are associated with this message.

Table 194 - Write Warning Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	4	Warning

8.3.6.3 Message: Hard Error

Owning Entity: SNIA

Message ID: SML3

Message Format String: The drive had a hard read or write error.

Table 195 describes the alerts that are associated with this message.

Table 195 - Hard Error Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	5	Warning

8.3.6.4 Message: Media

Owning Entity: SNIA

Message ID: SML4

Message Format String: Media can no longer be written/read, or performance is severely degraded.

Table 196 describes the alerts that are associated with this message.

Table 196 - Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.5 Message: Read Failure

Owning Entity: SNIA

Message ID: SML5

Message Format String: The drive can no longer read data from the storage media.

Table 197 describes the alerts that are associated with this message.

Table 197 - Read Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.6 Message: Write Failure

Owning Entity: SNIA

Message ID: SML6

Message Format String: The drive can no longer write data to the media.

Message: Media Life

Table 198 describes the alerts that are associated with this message.

Table 198 - Write Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.7 Message: Media Life

Owning Entity: SNIA

Message ID: SML7

Message Format String: The media has exceeded its specified life.

Table 199 describes the alerts that are associated with this message.

Table 199 - Media Life Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.8 Message: Not Data Grade

Owning Entity: SNIA

Message ID: SML8

Message Format String: The cartridge is not data-grade. Any data you write to the media is at risk. Replace the cartridge with a data-grade media.

Table 200 describes the alerts that are associated with this message.

Table 200 - Not Data Grade Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.9 Message: Write Protect

Owning Entity: SNIA

Message ID: SML9

Message Format String: Write command is attempted to a write protected media.

Table 201 describes the alerts that are associated with this message.

Table 201 - Write Protect Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.10 Message: No Removal

Owning Entity: SNIA

Message ID: SML10

Message Format String: Manual or software unload attempted when prevent media removal is on.

Table 202 describes the alerts that are associated with this message.

Table 202 - No Removal Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.11 Message: Cleaning Media

Owning Entity: SNIA

Message ID: SML11

Message Format String: Cleaning media loaded into drive

Table 203 describes the alerts that are associated with this message.

Table 203 - Cleaning Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.12 Message: Unsupported Format

Owning Entity: SNIA

Message ID: SML12

Message Format String: Attempted load of unsupported media format (e.g., DDS2 in DDS1 drive).

Message: Recoverable Snapped Tape

Table 204 describes the alerts that are associated with this message.

Table 204 - Unsupported Format Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.13 Message: Recoverable Snapped Tape

Owning Entity: SNIA

Message ID: SML13

Message Format String: Tape snapped/cut in the drive where media can be de-mounted.

Table 205 describes the alerts that are associated with this message.

Table 205 - Recoverable Snapped Tape Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.14 Message: Unrecoverable Snapped Tape

Owning Entity: SNIA

Message ID: SML14

Message Format String: Tape snapped/cut in the drive where media cannot be de-mounted.

Table 206 describes the alerts that are associated with this message.

Table 206 - Unrecoverable Snapped Tape Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.15 Message: Memory Chip In Cartridge Failure

Owning Entity: SNIA

Message ID: SML15

Message Format String: Memory chip failed in cartridge.

Table 207 describes the alerts that are associated with this message.

Table 207 - Memory Chip In Cartridge Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.16 Message: Forced Eject

Owning Entity: SNIA

Message ID: SML16

Message Format String: Manual or forced eject while drive actively writing or reading.

Table 208 describes the alerts that are associated with this message.

Table 208 - Forced Eject Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.17 Message: Read Only Format

Owning Entity: SNIA

Message ID: SML17

Message Format String: Media loaded that is read-only format.

Table 209 describes the alerts that are associated with this message.

Table 209 - Read Only Format Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.18 Message: Directory Corrupted On Load

Owning Entity: SNIA

Message ID: SML18

Message Format String: Drive powered down while loaded, or permanent error prevented the directory being updated.

Message: Nearing Media Life

Table 210 describes the alerts that are associated with this message.

Table 210 - Directory Corrupted On Load Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.19 Message: Nearing Media Life

Owning Entity: SNIA

Message ID: SML19

Message Format String: Media may have exceeded its specified number of passes.

Table 211 describes the alerts that are associated with this message.

Table 211 - Nearing Media Life Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.20 Message: Clean Now

Owning Entity: SNIA

Message ID: SML20

Message Format String: The drive thinks it has a head clog or needs cleaning.

Table 212 describes the alerts that are associated with this message.

Table 212 - Clean Now Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.21 Message: Clean Periodic

Owning Entity: SNIA

Message ID: SML21

Message Format String: The drive is ready for a periodic cleaning.

Message: Expired Cleaning Media

Table 213 describes the alerts that are associated with this message.

Table 213 - Clean Periodic Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.22 Message: Expired Cleaning Media

Owning Entity: SNIA

Message ID: SML22

Message Format String: The cleaning media has expired.

Table 214 describes the alerts that are associated with this message.

Table 214 - Expired Cleaning Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.23 Message: Invalid Cleaning Media

Owning Entity: SNIA

Message ID: SML23

Message Format String: Invalid cleaning media type used.

Table 215 describes the alerts that are associated with this message.

Table 215 - Invalid Cleaning Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.24 Message: Retention Requested

Owning Entity: SNIA

Message ID: SML24

Message Format String: The drive is having severe trouble reading or writing, which will be resolved by a retention cycle.

Message: Dual-Port Interface Error

Table 216 describes the alerts that are associated with this message.

Table 216 - Retention Requested Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Information

8.3.6.25 Message: Dual-Port Interface Error

Owning Entity: SNIA

Message ID: SML25

Message Format String: Failure of one interface port in a dual-port configuration (i.e., Fibre Channel)

Table 217 describes the alerts that are associated with this message.

Table 217 - Dual-Port Interface Error Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.26 Message: Drive Maintenance

Owning Entity: SNIA

Message ID: SML26

Message Format String: The drive requires preventive maintenance (not cleaning).

Table 218 describes the alerts that are associated with this message.

Table 218 - Drive Maintenance Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.27 Message: Hardware A

Owning Entity: SNIA

Message ID: SML27

Message Format String: The drive has a hardware fault that requires reset to recover.

Table 219 describes the alerts that are associated with this message.

Table 219 - Hardware A Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.28 Message: Hardware B

Owning Entity: SNIA

Message ID: SML28

Message Format String: The drive has a hardware fault that is not read/write related or requires a power cycle to recover.

Table 220 describes the alerts that are associated with this message.

Table 220 - Hardware B Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.29 Message: Interface

Owning Entity: SNIA

Message ID: SML29

Message Format String: The drive has identified an interface fault.

Table 221 describes the alerts that are associated with this message.

Table 221 - Interface Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.30 Message: Eject Media

Owning Entity: SNIA

Message ID: SML30

Message Format String: Error recovery action: Media Ejected

Message: Download Failure

Table 222 describes the alerts that are associated with this message.

Table 222 - Eject Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.31 Message: Download Failure

Owning Entity: SNIA

Message ID: SML31

Message Format String: Firmware download failed.

Table 223 describes the alerts that are associated with this message.

Table 223 - Download Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degraded/Warning

8.3.6.32 Message: Loader Hardware A

Owning Entity: SNIA

Message ID: SML32

Message Format String: Loader mechanism is having trouble communicating with the drive.

Table 224 describes the alerts that are associated with this message.

Table 224 - Loader Hardware A Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.33 Message: Loader Stray Media

Owning Entity: SNIA

Message ID: SML33

Message Format String: Stray media left in loader after previous error recovery.

Table 225 describes the alerts that are associated with this message.

Table 225 - Loader Stray Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.34 Message: Loader Hardware B

Owning Entity: SNIA

Message ID: SML34

Message Format String: Loader mechanism has a hardware fault.

Table 226 describes the alerts that are associated with this message.

Table 226 - Loader Hardware B Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.35 Message: Loader Door

Owning Entity: SNIA

Message ID: SML35

Message Format String: Changer door open.

Table 227 describes the alerts that are associated with this message.

Table 227 - Loader Door Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.36 Message: Loader Hardware C

Owning Entity: SNIA

Message ID: SML36

Message Format String: The loader mechanism has a hardware fault that is not mechanically related.

Message: Loader Magazine

Table 228 describes the alerts that are associated with this message.

Table 228 - Loader Hardware C Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.37 Message: Loader Magazine

Owning Entity: SNIA

Message ID: SML37

Message Format String: Loader magazine not present.

Table 229 describes the alerts that are associated with this message.

Table 229 - Loader Magazine Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.38 Message: Loader Predictive Failure

Owning Entity: SNIA

Message ID: SML38

Message Format String: Predictive failure of loader mechanism hardware

Table 230 describes the alerts that are associated with this message.

Table 230 - Loader Predictive Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.39 Message: Load Statistics

Owning Entity: SNIA

Message ID: SML39

Message Format String: Drive or library powered down with media loaded.

Table 231 describes the alerts that are associated with this message.

Table 231 - Load Statistics Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice or CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.40 Message: Media Directory Invalid at Unload

Owning Entity: SNIA

Message ID: SML40

Message Format String: Error preventing the media directory being updated on unload.

Table 232 describes the alerts that are associated with this message.

Table 232 - Media Directory Invalid at Unload Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.41 Message: Media System area Write Failure

Owning Entity: SNIA

Message ID: SML41

Message Format String: Write errors while writing the system area on unload.

Table 233 describes the alerts that are associated with this message.

Table 233 - Media System area Write Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.42 Message: Media System Area Read Failure

Owning Entity: SNIA

Message ID: SML42

Message Format String: Read errors while reading the system area on load.

Message: No Start of Data

Table 234 describes the alerts that are associated with this message.

Table 234 - Media System Area Read Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.43 Message: No Start of Data

Owning Entity: SNIA

Message ID: SML43

Message Format String: Media damaged, bulk erased, or incorrect format.

Table 235 describes the alerts that are associated with this message.

Table 235 - No Start of Data Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.44 Message: Loading Failure

Owning Entity: SNIA

Message ID: SML44

Message Format String: The drive is unable to load the media

Table 236 describes the alerts that are associated with this message.

Table 236 - Loading Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_MediaAccessDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.45 Message: Library Hardware A

Owning Entity: SNIA

Message ID: SML45

Message Format String: Changer mechanism is having trouble communicating with the internal drive

Table 237 describes the alerts that are associated with this message.

Table 237 - Library Hardware A Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.46 Message: Library Hardware B

Owning Entity: SNIA

Message ID: SML46

Message Format String: Changer mechanism has a hardware fault

Table 238 describes the alerts that are associated with this message.

Table 238 - Library Hardware B Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.47 Message: Library Hardware C

Owning Entity: SNIA

Message ID: SML47

Message Format String: The changer mechanism has a hardware fault that requires a reset to recover.

Table 239 describes the alerts that are associated with this message.

Table 239 - Library Hardware C Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.48 Message: Library Hardware D

Owning Entity: SNIA

Message ID: SML48

Message Format String: The changer mechanism has a hardware fault that is not mechanically related or requires a power cycle to recover.

Message: Library Diagnostic Required

Table 240 describes the alerts that are associated with this message.

Table 240 - Library Hardware D Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.49 Message: Library Diagnostic Required

Owning Entity: SNIA

Message ID: SML49

Message Format String: The changer mechanism may have a hardware fault which would be identified by extended diagnostics.

Table 241 describes the alerts that are associated with this message.

Table 241 - Library Diagnostic Required Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.50 Message: Library Interface

Owning Entity: SNIA

Message ID: SML50

Message Format String: The library has identified an interface fault

Table 242 describes the alerts that are associated with this message.

Table 242 - Library Interface Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.51 Message: Failure Prediction

Owning Entity: SNIA

Message ID: SML51

Message Format String: Predictive failure of library hardware

Table 243 describes the alerts that are associated with this message.

Table 243 - Failure Prediction Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ChangerDevice
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.52 Message: Library Maintenance

Owning Entity: SNIA

Message ID: SML52

Message Format String: Library preventative maintenance required.

Table 244 describes the alerts that are associated with this message.

Table 244 - Library Maintenance Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.53 Message: Library Humidity Limits

Owning Entity: SNIA

Message ID: SML53

Message Format String: Library humidity limits exceeded

Table 245 describes the alerts that are associated with this message.

Table 245 - Library Humidity Limits Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.54 Message: Library Voltage Limits

Owning Entity: SNIA

Message ID: SML54

Message Format String: Library voltage limits exceeded

Table 246 describes the alerts that are associated with this message.

Table 246 - Library Voltage Limits Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.55 Message: Library Stray Media

Owning Entity: SNIA

Message ID: SML55

Message Format String: Stray cartridge left in library after previous error recovery

Table 247 describes the alerts that are associated with this message.

Table 247 - Library Stray Media Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.56 Message: Library Pick Retry

Owning Entity: SNIA

Message ID: SML56

Message Format String: Operation to pick a cartridge from a slot had to perform an excessive number of retries before succeeding

Table 248 describes the alerts that are associated with this message.

Table 248 - Library Pick Retry Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.57 Message: Library Place Retry

Owning Entity: SNIA

Message ID: SML57

Message Format String: Operation to place a cartridge in a slot had to perform an excessive number of retries before succeeding

Table 249 describes the alerts that are associated with this message.

Table 249 - Library Place Retry Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.58 Message: Library Load Retry

Owning Entity: SNIA

Message ID: SML58

Message Format String: Operation to load a cartridge in a drive had to perform an excessive number of retries before succeeding

Table 250 describes the alerts that are associated with this message.

Table 250 - Library Load Retry Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.59 Message: Library Door

Owning Entity: SNIA

Message ID: SML59

Message Format String: Library door open is preventing the library from functioning

Table 251 describes the alerts that are associated with this message.

Table 251 - Library Door Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.60 Message: Library Mailslot

Owning Entity: SNIA

Message ID: SML60

Message Format String: Mechanical problem with import/export mailslot

Message: Library Magazine

Table 252 describes the alerts that are associated with this message.

Table 252 - Library Mailslot Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.61 Message: Library Magazine

Owning Entity: SNIA

Message ID: SML61

Message Format String: Library magazine not present

Table 253 describes the alerts that are associated with this message.

Table 253 - Library Magazine Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.62 Message: Library Security

Owning Entity: SNIA

Message ID: SML62

Message Format String: Library door opened then closed during operation

Table 254 describes the alerts that are associated with this message.

Table 254 - Library Security Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.63 Message: Library Security Mode

Owning Entity: SNIA

Message ID: SML63

Message Format String: Library security mode changed

Table 255 describes the alerts that are associated with this message.

Table 255 - Library Security Mode Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.64 Message: Library Offline

Owning Entity: SNIA

Message ID: SML64

Message Format String: Library manually turned offline

Table 256 describes the alerts that are associated with this message.

Table 256 - Library Offline Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.65 Message: Library Drive Offline

Owning Entity: SNIA

Message ID: SML65

Message Format String: Library turned internal drive offline.

Table 257 describes the alerts that are associated with this message.

Table 257 - Library Drive Offline Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

8.3.6.66 Message: Library Scan Retry

Owning Entity: SNIA

Message ID: SML66

Message Format String: Operation to scan the bar code on a cartridge had to perform an excessive number of retries before succeeding

Message: Library Inventory

Table 258 describes the alerts that are associated with this message.

Table 258 - Library Scan Retry Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.67 Message: Library Inventory

Owning Entity: SNIA

Message ID: SML67

Message Format String: Inconsistent media inventory

Table 259 describes the alerts that are associated with this message.

Table 259 - Library Inventory Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.68 Message: Library Illegal Operation

Owning Entity: SNIA

Message ID: SML68

Message Format String: Illegal operation detected

Table 260 describes the alerts that are associated with this message.

Table 260 - Library Illegal Operation Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Degrading/Warning

8.3.6.69 Message: Pass Through Mechanism Failure

Owning Entity: SNIA

Message ID: SML69

Message Format String: Error occurred in pass-through mechanism during self test or while attempting to transfer a cartridge between library modules

Table 261 describes the alerts that are associated with this message.

Table 261 - Pass Through Mechanism Failure Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.70 Message: Cartridge in Pass-through Mechanism

Owning Entity: SNIA

Message ID: SML70

Message Format String: Cartridge left in the pass-through mechanism between two library modules

Table 262 describes the alerts that are associated with this message.

Table 262 - Cartridge in Pass-through Mechanism Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

8.3.6.71 Message: Unreadable barcode Labels

Owning Entity: SNIA

Message ID: SML71

Message Format String: Unable to read a bar code label on a cartridge during library inventory/scan

Table 263 describes the alerts that are associated with this message.

Table 263 - Unreadable barcode Labels Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	2	Information

EXPERIMENTAL

8.3.6.72 Message: Throughput Threshold Warning Alert

Owning Entity: SNIA

Message ID: SML72

Message Format String: The throughput threshold has exceeded the warning level <ThroughputWarningAlertThreshold> of the <Computer System> system

Message: Throughput Threshold Critical Alert

Table 264 describes the message arguments.

Table 264 - Throughput Threshold Warning Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
ThroughputWarningAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.ThroughputWarningAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 265 describes the alerts that are associated with this message.

Table 265 - Throughput Threshold Warning Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.6.73 Message: Throughput Threshold Critical Alert

Owning Entity: SNIA

Message ID: SML73

Message Format String: The throughput threshold has exceeded the critical level <ThroughputCriticalAlertThreshold> of the <Computer System> system

Table 266 describes the message arguments.

Table 266 - Throughput Threshold Critical Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
ThroughputCriticalAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.ThroughputCriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 267 describes the alerts that are associated with this message.

Table 267 - Throughput Threshold Critical Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.6.74 Message: Physical Capacity Threshold Warning Alert

Owning Entity: SNIA

Message ID: SML74

Message Format String: The physical capacity threshold has exceeded the warning level <PhysicalCapacityWarningAlertThreshold> of the <Computer System> system

Table 268 describes the message arguments.

Table 268 - Physical Capacity Threshold Warning Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
PhysicalCapacityWarningAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.PhysicalCapacityWarningAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 269 describes the alerts that are associated with this message.

Table 269 - Physical Capacity Threshold Warning Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.6.75 Message: Physical Capacity Threshold Critical Alert

Owning Entity: SNIA

Message ID: SML75

Message Format String: The physical capacity threshold has exceeded the critical level <PhysicalCapacityCriticalAlertThreshold> of the <Computer System> system

Table 270 describes the message arguments.

Table 270 - Physical Capacity Threshold Critical Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
PhysicalCapacityCriticalAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.PhysicalCapacityCriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 271 describes the alerts that are associated with this message.

Table 271 - Physical Capacity Threshold Critical Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.6.76 Message: Logical Capacity Threshold Warning Alert

Owning Entity: SNIA

Message ID: SML76

Message Format String: The logical capacity threshold has exceeded the warning level <LogicalCapacityWarningAlertThreshold> of the <Computer System> system

Table 272 describes the message arguments.

Table 272 - Logical Capacity Threshold Warning Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
LogicalCapacityWarningAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.LogicalCapacityWarningAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 273 describes the alerts that are associated with this message.

Table 273 - Logical Capacity Threshold Warning Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.6.77 Message: Logical Capacity Threshold Critical Alert

Owning Entity: SNIA

Message ID: SML77

Message Format String: The logical capacity threshold has exceeded the critical level <LogicalCapacityCriticalAlertThreshold> of the <Computer System> system

Table 274 describes the message arguments.

Table 274 - Logical Capacity Threshold Critical Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
LogicalCapacityCriticalAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.LogicalCapacityCriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Message: System Ratio Threshold Warning Alert

Table 275 describes the alerts that are associated with this message.

Table 275 - Logical Capacity Threshold Critical Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.6.78 Message: System Ratio Threshold Warning Alert

Owning Entity: SNIA

Message ID: SML78

Message Format String: The system ratio has fallen below the warning level threshold <SystemRatioWarningAlertThreshold> of the <Computer System> system

Table 276 describes the message arguments.

Table 276 - System Ratio Threshold Warning Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
SystemRatioWarningAlert Threshold	string	A string rendering of the CIM_VTLResourceUsage.LogicalCapacity CriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System The Name of a Virtual Tape Library System

Table 277 describes the alerts that are associated with this message.

Table 277 - System Ratio Threshold Warning Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.6.79 Message: System Ratio Threshold Critical Alert

Owning Entity: SNIA

Message ID: SML79

Message Format String: The system ratio threshold has fallen below the critical level threshold <SystemRatioCriticalAlertThreshold> of the <Computer System> system

Table 278 describes the message arguments.

Table 278 - System Ratio Threshold Critical Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
SystemRatioCriticalAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.SystemRatioCriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 279 describes the alerts that are associated with this message.

Table 279 - System Ratio Threshold Critical Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.6.80 Message: Deduplication Ratio Threshold Warning Alert

Owning Entity: SNIA

Message ID: SML80

Message Format String: The deduplication ratio has fallen below the warning level threshold <DeduplicationRatioWarningAlertThreshold> of the <Computer System> system

Message: Deduplication Ratio Threshold Critical Alert

Table 280 describes the message arguments.

Table 280 - Deduplication Ratio Threshold Warning Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
DeduplicationRatioWarningAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.DeduplicationRatioWarningAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 281 describes the alerts that are associated with this message.

Table 281 - Deduplication Ratio Threshold Warning Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.6.81 Message: Deduplication Ratio Threshold Critical Alert

Owning Entity: SNIA

Message ID: SML81

Message Format String: The deduplication ratio threshold has fallen below the critical level threshold <DeduplicationRatioCriticalAlertThreshold> of the <Computer System> system

Table 282 describes the message arguments.

Table 282 - Deduplication Ratio Threshold Critical Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
DeduplicationRatioCriticalAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.DeduplicationRatioCriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 283 describes the alerts that are associated with this message.

Table 283 - Deduplication Ratio Threshold Critical Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

EXPERIMENTAL

8.3.6.82 Message: Replication Traffic Threshold Warning Alert

Owning Entity: SNIA

Message ID: SML82

Message Format String: The replication traffic threshold has exceeded the warning level <ReplicationTrafficWarningAlertThreshold> of the <Computer System> system

Table 284 describes the message arguments.

Table 284 - Replication Traffic Threshold Warning Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
ReplicationTrafficWarningAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.ReplicationTrafficWarningAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System The Name of a Virtual Tape Library System

Table 285 describes the alerts that are associated with this message.

Table 285 - Replication Traffic Threshold Warning Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	3	Warning

EXPERIMENTAL

EXPERIMENTAL

8.3.6.83 Message: Replication Traffic Threshold Critical Alert

Owning Entity: SNIA

Message ID: SML83

Message Format String: The replication traffic threshold has exceeded the critical level <ReplicationTrafficCriticalAlertThreshold> of the <Computer System> system

Table 286 describes the message arguments.

Table 286 - Replication Traffic Threshold Critical Alert Message Arguments

Message Argument	Data Type	Description	Possible Values
ReplicationTrafficCriticalAlertThreshold	string	A string rendering of the CIM_VTLResourceUsage.ReplicationTrafficCriticalAlertThreshold property.	A number between 0 and 1
Computer System	string	The Name property of the Virtual Library System or Virtual Tape Library	The Name of a Virtual Library System
			The Name of a Virtual Tape Library System

Table 287 describes the alerts that are associated with this message.

Table 287 - Replication Traffic Threshold Critical Alert Alert Information

Name	Req	Value	Description
ALERTING_MANAGED_ELEMENT	Y		CIM_ComputerSystem
ALERT_TYPE	Y		Device Alert
PERCEIVED_SEVERITY	Y	6	Critical

EXPERIMENTAL

9 Service Discovery

9.1 Objectives

Service discovery in the context of SMI-S refers to the discovery of dedicated SMI-S servers, general purpose SMI-S servers, and directory servers, and the functions they offer in an SMI-S managed environment. The specific objectives to be addressed by the discovery architecture are:

- 1) Provide a mechanism that allows SMI-S clients to discover the SMI-S constituents in a storage network environment so that they may communicate with these constituents using CIM Operations over HTTP protocol. This includes:
 - Finding the address for the SMI-S constituent;
 - Finding the capabilities of the server, including communications capabilities, security capabilities, CIM operational capabilities and the functional capabilities (CQL, Batch operations support, etc.);
- 2) Provide a mechanism that is efficient in the amount of information exchanged with minimal exchanges to acquire the information;
- 3) Provide a mechanism that accurately defines the services in the network, independent of whether or not those services are currently available;
- 4) Provide a mechanism that provides information on namespaces provided and the CIM Schema supported;
- 5) Provide a mechanism that allows SMI-S clients the profile(s) supported by agents and object managers;
- 6) Provide a mechanism that scales to enterprise environments;
- 7) Utilize existing standard mechanisms to effect the SMI-S service discovery to enable rapid deployment;
- 8) Provide a mechanism that allows SMI-S clients to determine the level of (SMI-S) support provided by the constituents (e.g., R1, R2, etc.)

9.2 Overview

SMI-S uses the Service Location Protocol Version 2 (SLPv2), as defined by IETF RFC 2608, for its *basic* discovery mechanism. SLPv2 is used to locate constituents (agents, object managers, etc.), but complete discovery of all the services offered involves traversing the interoperability model for the SMI-S profile supported. This clause of the SMI-S specification deals primarily with the information discovered using SLPv2. There are references to information discovered by traversing the interoperability model, but details on this are provided in 11.3.

NOTE SLPv1 is not supported in SMI-S as discovery mechanism. SMI-S requires capabilities that were introduced in SLPv2 in order to support the discovery of SMI-S agents and object managers.

SLPv2 defines discovery protocols among three constituents:

User Agent (UA): A process that attempts to establish contact with one or more services. A User Agent retrieves service information from Service Agents or Directory Agents. In SMI-S, a “user agent” would be part of an SMI-S Client.

Service Agent (SA): A process working on behalf of one or more services to advertise the services. In SMI-S, a “service agent” would be supported by SMI-S dedicated or general purpose servers.

Directory Agent (DA): A process that caches SLP service advertisements registered by Service Agents and forwards the service advertisements to User Agents on demand. In SMI-S, the SLP “Directory agent” is defined as the main function of the “directory server” role in the SMI-S Reference Model. SMI-S allows

multiple Directory Agents to be used for purposes including load sharing and availability. These Directory Agents may have the same scope, as allowed by SLPv2.

SLPv2 provides a framework for client applications, represented by User Agents, to find and utilize services, represented by Service Agents. The Directory Agents represent an optional part that enhances the performance and scalability of the protocol by acting as a cache for all services that have been advertised. Directory Agents also reduce the load on Service Agents, making simpler implementations of Service Agents possible. User Agents can then query the Directory Agents for services. Service Agents register with Directory Agents and are required to re-register as the registrations expire. If no Directory Agents are present, User Agents may request service information directly from the Service Agents.

Using SLPv2, a client can discover SMI-S servers and SLPv2 Directory Agents in the storage network. In the case of SMI-S servers, the basic information discovered is the profiles supported and the URL of the service. Details on the specific services provided with the profile are then found by traversing the service structure modeled for the profile.

Using SLPv2, a “service agent” advertises its services. These advertisements have an expiration time period. To avoid getting an advertisement deleted, a service agent shall reregister before the time period expires. SMI-S servers may deregister as part of a graceful shutdown.

A service advertisement consists of file components:

- Service type name – describes the general type of service being advertised (ex. Printing, faxing, etc.). The working assumption is that DMTF wants “WBEM Servers” advertised with the service type WBEM. This is used by SMI-S servers (both dedicated and general purpose servers).
- Attributes – The collection of attributes describes the particular instance of the service in more detail. For SMI-S, these would be the attributes defined by the service type template for WBEM. The attributes are defined in 9.5.2.
- Service Access point – the service access point defines the point of connection that the software client of the UA uses to connect to the service over the network.
- Scopes – These are administrative groupings of services. The default value (“default”) should be used for SMI-S servers. Other scopes may be defined by the customer, but care must be taken when this is done. The administrator shall do this correctly or SMI-S servers will not be visible.
- Language – Services advertisements contain human readable strings. These are provided in English, but may also be in other languages.

IMPLEMENTED

SLPv2 provides for authentication of service URLs and service attributes. This provides user agents (UAs) and directory agents (DAs) with assurances of the integrity of service URLs and attributes included in SLP messages. The only systems which can generate digital signatures are those which have been configured by administrators in advance. Agents that verify signed data may assume it is trustworthy inasmuch as administrators have assured trustworthiness through the cryptographic keying of SAs and DAs. The SLPv2 security model assumes that service information is public, and therefore does not require confidentiality.

Clause 2.5 of RFC 3723, *Securing Block Storage Protocols over IP*, states that the SA advertisements as well as UA requests and/or responses are vulnerable to these security threats:

- 1) An attacker could insert or alter service agent (SA) advertisements or responses to a UA requests in order to masquerade as the real peer or launch a denial of service attack.

- 2) An attacker could gain knowledge about an SA or a UA through sniffing, and launch an attack against the peer.
- 3) An attacker could spoof DA advertisements and thereby cause UAs and SAs to use a rogue DA.

Clause 2.5 of RFC 3723 also outlines the capabilities required to address these threats, but notes that SLP (as defined in RFC 2608) does not satisfy these security requirements. SLPv2 only provides end-to-end authentication (i.e., does not support confidentiality), but with this authentication, there is no way to authenticate zero result responses. Thus an attacker could mount a denial of service attack by sending UAs a zero results Service Reply (SrvRply) or Attribute Reply (AttrRply) with a source address corresponding to a legitimate DA advertisement.

The RFC 3723 mitigation strategies include reliance on digital signatures for authentication of service URLs and attributes as well as IPsec. For SMI-S environments that require security in conjunction with the use of SLPv2, the major RFC 3723 recommendations are not necessary as long as the SLP messages are not fully trusted and SSL or TLS with server certificates are used. Additional security guidance is provided in the sections associated with UAs and SAs.

IMPLEMENTED

9.3 SLP Messages

SLP v2 divides the base set of SLP messages into required and optional subsets.

NOTE SLP v2 also includes a new feature, an extension format. Extension messages are attached to base messages. SMI-S does not use extensions. The discussion of messages introduces terms that define the SLP services:

- Attribute Reply (AttrRply): A reply to an Attribute Request. (optional)
- Attribute Request (AttrRqst): A request for attributes of a given type of service or attributes of a given service. (optional)
- DA Advertisements (DAAdvert): A solicited (unicast) or unsolicited (multicast) advertisement of Directory Agent availability.
- SA Advertisement (SAAdvert): Information describing a service that consists of the Service Type, Service Access Point, lifetime, and Attributes.
- Service Acknowledgement (SrvAck): A reply to a SrvReg request.
- Service Deregister (SrvDereg): A request to deregister a service or some attributes of a service. (optional)
- Service Register (SrvReg): A request to register a service or some attributes of a service.
- Service Reply (SrvRply): A reply to a Service Request.
- Service Request (SrvRqst): A request for a service on the network.
- Service Type Reply (SrvTypeRply): A reply to a Service Type Request. (optional)
- Service Type Request (SrvTypeRqst): A request for all types of service on the network. (optional)

Service Agents (SAs) and User Agents (UAs) shall support Service Request, Service Reply, and DAAdvertisement message types. Service Agents shall additionally support Service Registration, SA Advertisement, and Service Acknowledgment message types. The remaining message types may be supported by Service Agents and User Agents. Directory Agents (DAs) shall support all message types

with the exception of SA Advertisement. Table 288 lists each base message type, its abbreviation, function code, and required/optional status.

Table 288 - Message Types

Message Type	Abbreviation	Function Code	Required (R)/ Optional (O)		
			DAs	SAs	UAs
Service Request	SrvRqst	1	R	R	R
Service Reply	SrvRply	2	R	R	R
Service Registration	SrvReg	3	R	R	O
Service Deregistration	SrvDereg	4	R	O	O
Service Acknowledgement	SrvAck	5	R	R	O
Attribute Request	AttrRqst	6	R	R	R
Attribute Reply	AttrRply	7	R	R	R
DA Advertisement	DAAadvert	8	R	R	R
Service Type Request	SrvTypeRqst	9	R	O	O
Service Type Reply	SrvTypeRply	10	R	O	O
SA Advertisement	SAAadvert	11	N/A	R	O

NOTE The requirements in Table 288 extend the requirements defined for SLP V2. SMI-S adds additional requirements for AttrRqst and AttrRply beyond those defined by the RFC.

9.4 Scopes

SLPv2 defines a scope as follows:

Scope: A set of services, typically making up a logical administrative group.

Scopes are sets of service instances. The primary use of Scopes is to provide the ability to create administrative groupings of services. A set of services may be assigned a scope by network administrators. A User Agent (UA) seeking services is configured to use one or more scopes. The UA only discovers those services that have been configured for it to use. By configuring UAs and Service Agents with scopes, administrators may make services available. Scopes strings are case insensitive. The default SCOPE string is "DEFAULT".

SMI-S does not dictate how Scopes are set. That is, scopes can be set by customers to match their needs. However, SMI-S requires that SMI-S servers use the "default" scope as a means of making SMI-S advertisements visible to SMI-S clients.

To be compliant with SMI-S, User Agents (SMI-S clients) and Service Agents (SMI-S servers) shall not require scope settings that interfere with administrative use of scopes. Specifically, this means:

- SMI-S clients and servers shall allow an administrator to set scopes to define what is to be searched, and,
- SMI-S clients and servers shall allow an administrator to configure scopes, including turning off the "default" scope.

9.5 Services Definition

Services definition uses these terms defined in SLPv2:

- **Service Type Template:** A formalized, computer-readable description of a Service Type. The template defines the format of the service URL and attributes supported by the service type.
- **Service URL:** A Uniform Resource Locator for a service containing the service type name, network family, Service Access Point, and any other information needed to contact the service.

Services are defined by two components: the Service URL and the Service Type Template. The Service URL defines an access point for the service and identifies a unique resource in the network. Service URLs may be either existing generic URLs or URLs from the service: URL scheme.

The second component in a Service definition is a Service Type Template. Service Type Templates define the attributes associated with a service. These attributes, through inclusion in registrations and queries, allow clients to differentiate between similar services.

SMI-S servers use a Service Type Template defined by DMTF for advertising "WBEM Servers". The template name for WBEM Servers is "WBEM".

9.5.1 Service Type

Service Type: The class of a network service represented by a unique string (for example a namespace assigned by IANA).

The service type describes a class of services that share the same attributes (e.g., the service printer or the service "WBEM").

The basic function of SLP discovery is the identification of the service offered by a constituent. In the case of SMI-S, the service type advertised by all constituents is "WBEM." This follows a DMTF proposal for advertising WBEM Servers. The only exception to this is the Directory Server, which advertises itself as a "directory-agent." That is, SMI-S uses a standard SLP directory service. SMI-S does not require a unique SMI-S directory server.

For other roles (SMI-S servers) the role advertises its services as a WBEM services (e.g., "WBEM").

9.5.2 Service Attributes

Attributes: A collection of tags and values describing the characteristics of a service.

The attributes are defined in the DMTF WBEM SLP Template (DSP0206) 1.0.

9.6 User Agents (UA)

A User Agent is a Client process working on the user's behalf to establish contact with some service. A User Agent retrieves service information from Service Agents (9.7) or Directory Agents (9.8). Further description of a Client and its role may be found in 11.2, "SMI-S Client".

The only required feature of a User Agent is that it can issue SrvRqsts and interpret DAAdverts, SAAAdverts and SrvRply messages. If Directory Agents exist, User Agents shall issue requests as Directory Agents are discovered.

An SMI-S Client should act as an SLP user agent (UA) using the query functions of SLP V2 to determine location and other attributes of the "WBEM" SLP Service Type Template defined in , "".

The basic search methodology for SMI-S clients is to search for directory agents and service agents within their scope. If all SMI-S servers are supported by a directory agent, then the search yields nothing but directory agents. The client can then obtain a list of services (and their URLs) for management of the SMI-S servers.

If any Service agents are not covered by a directory agent (i.e., are not within its scope), then the client obtains service replies from those service agents.

An client would typically search for all service types available in their scope(s). This returns a list of service types available in the network. However, an SMI-S client can be assumed to be searching for “WBEM” service types. If a client only manages selected devices (e.g., switches or arrays), the SMI-S client can issue a request for the specific services by using predicates on the “RegisteredProfilesSupported” attribute.

IMPLEMENTED

When a SMI-S client uses SLPv2 and security is an issue, the following should be considered:

- SSL and TLS should be used with a certificate-based cipher suite along with a certificate installed on each SMI-S server (SA) for communications with discovered SAs (SMI-S servers).
- SLPv2 Service Agents (SA) and Directory Agents (DA) may advertise (SAA adverts and DAA adverts, respectively) their presence on the network, using multicast; however, SMI-S clients should treat these advertisements as advisory (i.e., identity shall be verified as described in 9.7 and 9.8).
- SMI-S clients should maintain and use a negative authentication cache to avoid repeatedly contacting an SMI-S server that fails to authenticate as part of the SSL or TLS handshake.

IMPLEMENTED

9.7 Service Agents (SAs)

A Service Agent supports an SMI-S server process working on behalf of one or more services to advertise the services.

See Clause 11 SMI-S Roles for further description of SMI-S servers.

Service Agents shall accept multicast service requests and unicast service requests. SAs may accept other requests (Attribute and Service Type Requests). An SA shall reply to appropriate SrvRqsts with SrvRply or SAAadvert messages. The SA shall also register with all DAs as they are discovered.

To provide for SMI-S Client discovery of SMI-S servers, a WBEM Server shall act as a Service agent (SA) for the IETF Service Level Protocol (SLP) V2 as defined in IETF RFC 2608. The service shall correspond to V2 of SLP (IETF RFC 2608 and 2609) and shall use the Service Templates defined in of this specification for advertisements. An SMI-S server acting as an SA shall provide a separate SLP advertisement for each address/port that the WBEM Server advertises.

IMPLEMENTED

When a SMI-S server uses SLPv2 and security is an issue, the following should be considered:

- SMI-S servers should accept SSL and TLS unicast connections from SMI-S clients as well as selecting a certificate-based cipher suite.
- SMI-S servers that advertise their existence as SLPv2 SAs (SAA adverts) should minimize leakage of information, by minimizing the information that is contained in the multicast advertisements.
- SMI-S servers, functioning as SAs, should register with all discovered DAs, which advertise any of its configured scopes and establish connections with these DAs over unicast.

- When SMI-S servers are also functioning as clients (e.g., cascading), they should follow the security guidance provided in 9.6 User Agents (UA).

IMPLEMENTED

9.8 Directory Agents (DAs)

SMI-S supports existing SLPv2 Directory Agents (without modification). That is, SMI-S makes no assumptions on Directory Agents that are not made by SLPv2. Note that this cannot quite be said for User Agents, which are looking for SMI-S specific services, or Service Agents, which are advertising SMI-S specific services.

9.9 Service Agent Server (SA Server)

9.9.1 General Information

The reserved listening port for SLP is 427, the destination port for all SLP messages. Service Agents (SAs) are required to listen for both unicast and multicast requests. A Directory Agent (DA) shall listen for unicast request and specific multicast DA discovery service requests. SAs and User Agents (UAs) that perform passive DA discovery shall listen for multicast DA Advertisements (DAA adverts).

TCP/IP requires that a single server process per network interface control all incoming messages to a port. That requirement necessitates a mechanism to share the SLP port (427).

Sharing the SLP port (427) is accomplished with a Service Agent Server (SA Server) process that 'owns' the port on behalf of all SAs, UAs and optional DA that are listening for SLP messages. The SA Server listens for incoming messages that request advertisement information and either answer each request or forward it to the appropriate SA. The SA Server also performs passive DA discovery and distribute the DA addresses and scopes to the SAs and UAs that it serves.

A SA Server may also function as a DA if the SA Server is implemented so that it answers requests for advertisement information rather than forwarding each request to the appropriate SA. The combined DA/SA Server is acting as an intermediary between a SA that registered an advertisement and a UA requesting information about the advertisement.

9.9.2 SA Server (SAS) Implementation

IETF RFC 2614 describes APIs for both the C and Java languages. Both APIs are designed for standardized access to the Service Location Protocol (SLP).

The goals of the C API are:

- Directly reflect the structure of SLP messages in API calls and return types as character buffers and other simple data structures.
- Simplify memory management to reduce API client requirements.
- Provide API coverage of just the SLP protocol operations to reduce complexity.
- Allow incremental and asynchronous access to return values, so small memory implementations are possible.
- Support multithreaded library calls on platforms where thread packages are available.

The Java API goals are:

- Provide complete coverage of all protocol features, including service type templates, through a programmatic interface.
- Encourage modularity so that implementations can omit parts of the protocol that are not needed.
- In conformance with Java's object-oriented nature, reflect the important SLP entities as objects and make the API itself object-oriented.
- Use flexible collection data types consistently in the API to simplify construction of parameters and analysis of results.
- Designed for small code size to help reduce download time in networked computers.

9.9.3 SA Server (SAS) Clients

9.9.3.1 Description

An SAS Client is a Service Agent (SA), User Agent (UA), or Directory Agent (DA) that is associated with a SA Server. The SA Server listens on the SLP port (427) and appropriately handle all incoming messages for each SAS Client. A DA acting as a SAS Client is separately configured on the same host as the SA Server.

9.9.3.2 SAS Client Requests – SA Server Responses

A SA Server responds when appropriate, to incoming unicast and multicast messages from SAS Clients. The SA Server may answer with the appropriate advertisement, if available, or forward the request on to the appropriate SAS Client. If the SA Server is also functioning as a DA, it discards a multicast SrvRqst of "service:directory-agent" that has either a missing scope list or the scope list does not contain a scope the Service Agent Server/DA is configured with.

9.9.4 SA Server Configuration

9.9.4.1 Overview

SA Servers may be configured via an individual SLP configuration file, programmatically, or a combination of the two. DHCP may also be used obtain the scope list for a SA Server. Figure 18 illustrates the various means of configuring a SA Server.

9.9.4.2 SLP Configuration File

If a SA Server is also functioning as a DA, the DA configuration properties shown in Table 289 shall be set:

Table 289 - Required Configuration Properties for SA as DA

Keyword	Data Type	Value
net.slp.isDA	boolean	true
net.slp.DAAttributes	string	(SA-Server=true)

The DA attribute/value pair of "SA-Server=true" allows a query to be used when a SA Server/DA needs to be identified. In addition, when the SA Server/DA responds to a SrvRqst message with a DAAdvert message, the DA attribute/value pair is included.

The remaining DA configuration property, net.slp.DAHeartBeat, with a default of 10,800 seconds, may be set as appropriate. If a SA Server is not functioning as a DA, the SA configuration property in Table 290 shall be set:

Table 290 - Required Configuration Properties for SA

Keyword	Data Type	Value
net.slp.SAAttributes	string	(SA-Server=true)

9.9.4.3 Programmatic Configuration

Both the C and Java language API's provide access to SLP properties contained in the SLP configuration file. The actual SLP configuration file is not accessed or modified via the interfaces. Once the file is loaded into memory at the start of execution, the configuration property accessors work on the in-memory representation.

The C language API provides the `SLPGetProperty()` and `SLPSetProperty()` functions. The `SLPGetProperty()` function allows read access to the SLP configuration properties while the `SLPSetProperty()` function allows modification of the configuration properties.

The `SLPSetProperty()` function has the following prototype:

```
void SLPSetProperty(const char *pcName, const char *pcValue);
```

The `SLPSetProperty()` function takes two string parameters: `pcName` and `pcValue`. The `pcName` parameter contains the property name and `pcValue` contains the property value. The following example uses the `SLPSetProperty()` function to configure a SA Server that is not functioning as a DA:

```
void setSAAttributes() {
    char value[80]; /* A buffer for storing the attribute string. */
    value = "SA Server=true";
    SLPSetProperty("net.slp.SAAttributes", value);
}
```

9.9.4.4 DHCP Configuration

If the Service Agent Server is also functioning as a DA, its scope list may be obtained via DHCP. Scopes discovered via DHCP take precedence over the `net.slp.useScopes` property in the SLP configuration file.

9.9.4.5 Scope

A Service Agent Server is configured with a minimum scope of DEFAULT. If a Service Agent Server is not functioning as a DA, DEFAULT is the only scope configured. If a Service Agent Server is functioning as a DA, it may have additional scopes configured. Use of the DEFAULT scope enables the associated SAS Clients (UAs, SAs and DA) to actively discover the Service Agent Server using a well-known value for scope.

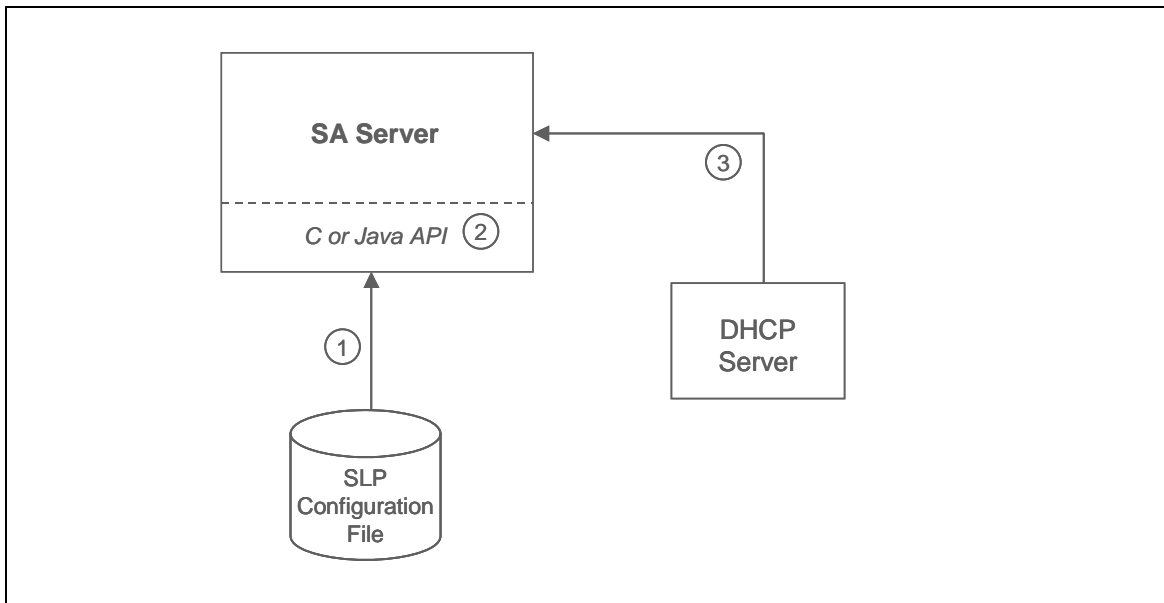


Figure 18 - SA Server Configuration

- 1) The SA Server may obtain specific configuration values via an individual SLP Configuration file.
- 2) The C or Java API provides programmatic access to the configuration file properties.
- 3) The SA Server may obtain its scope values from a DHCP Server.

9.9.5 SA Server Discovery

“Discovery” of a SA Server by its SAS Clients is accomplished by successfully establishing the required communication link between the two entities. There is no need for active or passive discovery as described by SLP since both the SA Server and SAS Clients reside on the same host system.

9.9.6 SAS Client Registration

Service Agents (SAs) that are SAS Clients register and deregister with the local SA Server using the SrvReg/SrvDereg messages. The SA Server responds with a Service Acknowledgement (SrvAck) message. The SA Server store a service advertisement until either its lifetime expires or a SrvDereg message is received.

If the SA Server is also functioning as a DA, the DA registration requirement is also met. The SA server also forwards any SA registration to other DAs that have the same scope as the SA.

9.10 Configurations

There are three network configurations (9.10.1, 9.10.2, 9.10.3) showing SMI-S clients and servers. The routing of SLP's multicast messages effect the SMI-S discovery process. SMI-S clients and servers shall be able to be configured to work in these environments.

9.10.1 Multicast Configurations

This is the simplest environment and is shown in Figure 19. This network allows multicast messages to be delivered to all the components of a SMI-S management system. As defined in IETF RFC 2608 - 8.1, the client uses multicast SLP messages to contact the SLP Service Agent (SA) associated with each SMI-S server. Then, each SA sends replies directly back to the client.

Because of the possible size of the reply, servers shall support TCP/IP (as well as UDP) to send the reply. The server shall also support the SLP oversize bit to tell the client large TCP/IP messages shall be used. When a client sees the overflow bit in a UDP response, it should retry using a TCP request.

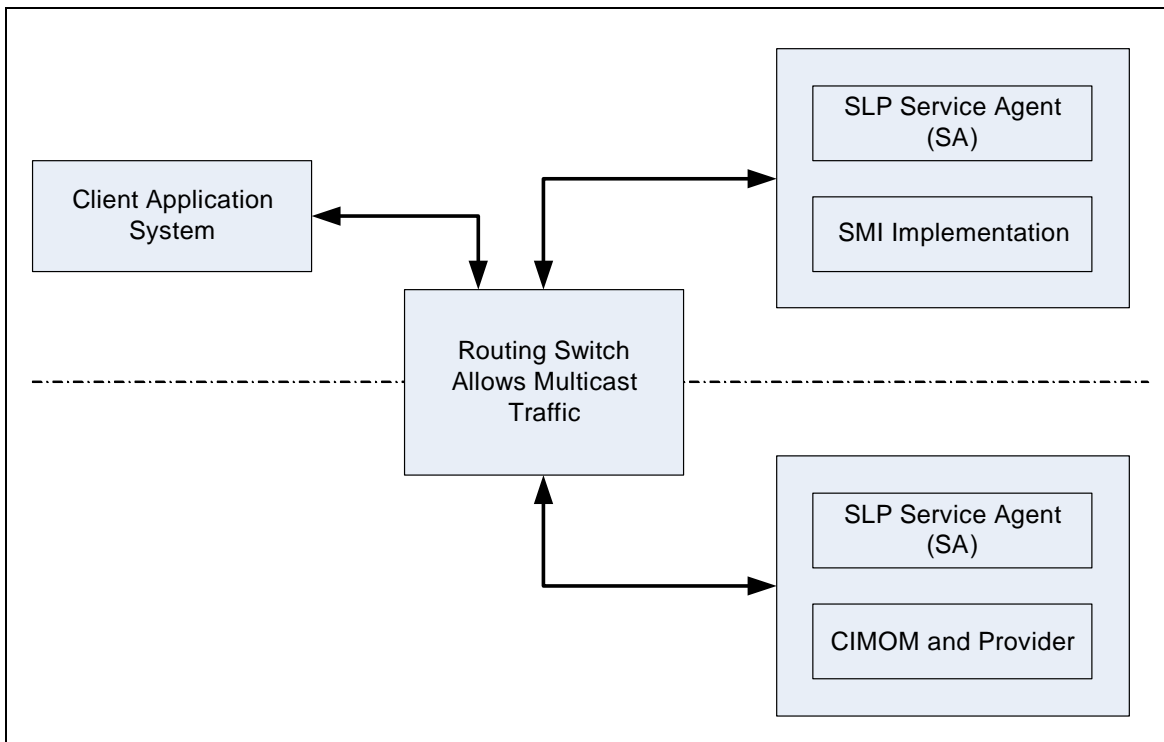


Figure 19 - Multicast Configuration

9.10.2 No Multicast configuration

In this configuration, shown in Figure 20, the network doesn't allow the use of multicast messages. All communication shall use TCP/IP point to point connections. First, a SLP directory agent should be used. Each SA shall be configurable by the user. The user will configure the SA by setting the address of the SLP directory agent (DA). At startup each SA shall use a temporary registration to tell the DA its SLP information (IETF RFC 2608 - 8.3). The SAs shall renew the registration before it expires (IETF RFC 2608 - 8.3). The registration timeout should be about 5-10 min.

Scope

The client shall also be configurable by the user. The user will configure the client by setting the DA address. The client will use this address to send SLP messages to the DA (IETF RFC 2608 - 8.1). The DA will satisfy the requests using information provided by the SAs.

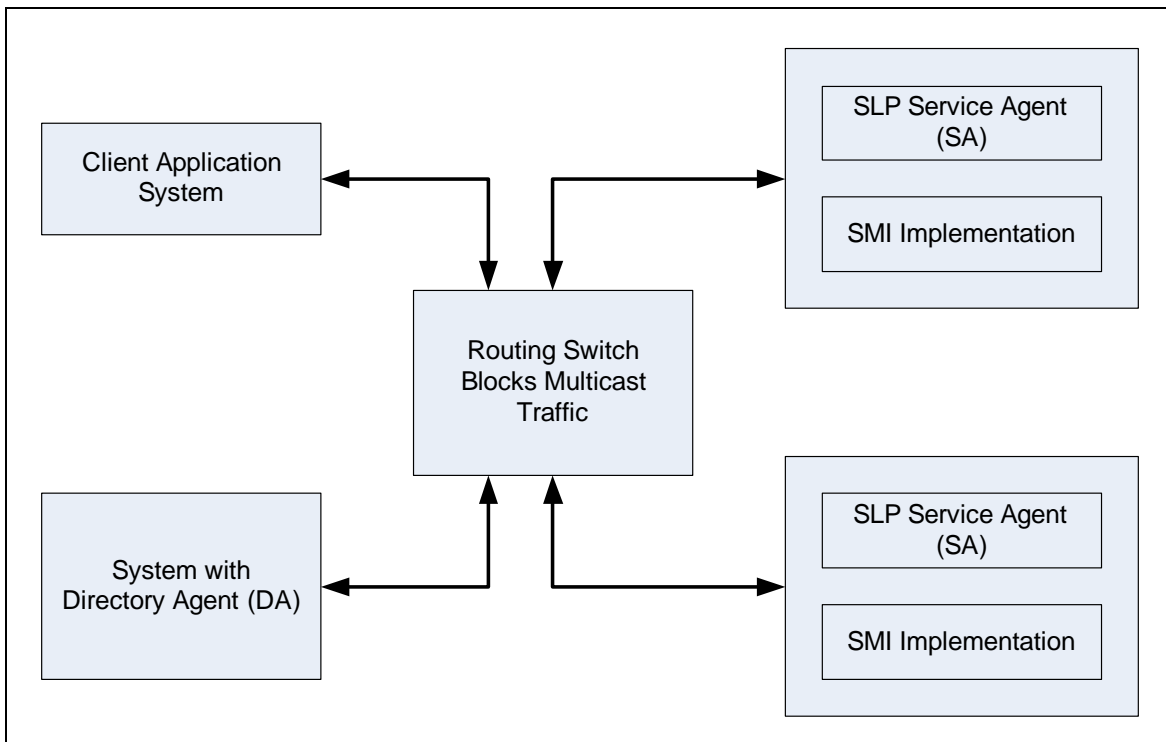


Figure 20 - No Multicast configuration

9.10.3 Multicast Islands

Networks that allow for multicast messages to reach parts of the system, require the use of both techniques described. The client should use the multicast process and the no multicast method. It should be able to combine the information found each way into a single set of discovery information. The SAs shall support both methods at the same time as shown in Figure 21.

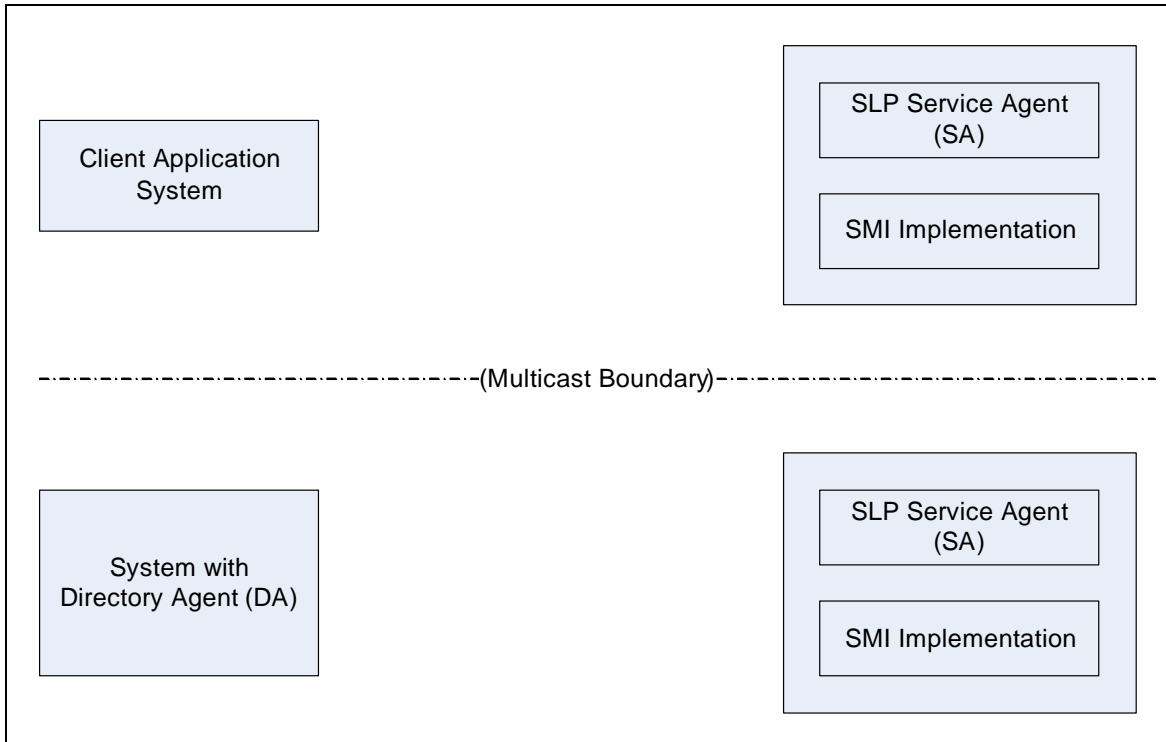


Figure 21 - Multicast Islands

Scope

10 Indications

10.1 Indications profile

10.1.1 Profile Support

In this version of the SMI-S specification, the only supported profile is the DMTF Indications Profile, DSP 1054, version 1.2.2.

10.1.2 Creating a client defined indication and subscription

Table 291 identifies the elements of the use case to create an indication filter and subscribe to it.

Table 291 - Create an IndicationFilter and subscribe to it

Use Case Element	Description
Summary	Given a top level system of an autonomous profile and a URI for an indication listener create a client defined indication and subscribe to it.
Basic Course of Events	1. Get the IndicationConfigurationService if it exists 1b. If not try to do a CreateInstance for the IndicationFilter (if it succeeds, then continue) 2. If the service exists, get the IndicationConfigurationCapabilities to find out if the implementation supports client defined IndicationFilters 3. If the capability exists, then do a CreateAndSubscribe for the indication 3b. Do a CreateInstance on the ListenerDestination and another CreateInstance on the IndicationSubscription
Exception Paths	None
Triggers	The administrator wants to listen for a specific indication of his/her choosing.
Assumptions	The implementation supports client defined IndicationFilters
Preconditions	The top level system of the profile and a listener destination for the application to get the indications.
Postconditions	The IndicationFilter is created and a subscription to it is recorded in the WBEM Server.

10.1.3 ListenerDestination

For CIM-XML (required by this version of the standard), Listener Destinations shall use the subclass CIM_ListenerDestinationCIMXML.

10.2 Indication Filter Strings

WBEM indications are defined using filter strings. The filter strings are expressed in a query language that includes the type of indication and related CIM elements.

For versions of this standard starting with 1.3.0, new indication filters shall be defined using CQL (see DMTF DSP0202). WQL is no longer supported in the current version of this standard.

Although CQL supports complex filter strings, the filters used in SMI-S are very simple and may be expressed as a few patterns – literal text containing a limited number of variables representing CIM elements. The patterns are defined in the following simple grammar:

- literal text does not include curly brackets (“{” and “}”)
- variables are surrounded by curly brackets; the usage of variables is explained in the “Semantic” sub-section following each filter string

10.2.1 Instance Creation

10.2.1.1 Filter String

```
SELECT * FROM CIM_InstCreation WHERE
    SourceInstance ISA {class-name}
```

10.2.1.2 Semantic

An instance of a class is instantiated. {class-name} is the name of a class (or one of its subclasses) of the instance created.

10.2.2 Instance Deletion

10.2.2.1 Filter String

```
SELECT * FROM CIM_InstDeletion WHERE
    SourceInstance ISA {class-name}
```

10.2.2.2 Semantic

An instance of a class is deleted. {class-name} is the name of a class (or one of its subclasses) of the instance deleted.

10.2.3 Modification of any value in an array property

10.2.3.1 Filter string

```
SELECT * FROM CIM_InstModification WHERE
    SourceInstance ISA {class-name} AND
    SourceInstance.{class-name}::{property-name} <>
    PreviousInstance.{class-name}::{property-name}
```

10.2.3.2 Semantic

One of the values of the array property {property-name} in class {class-name} (or one of its subclasses) has been modified, or an additional value is added to {property-name} or a value is removed from {property-name}.

10.2.4 Modification to either of Two Specific values in an Array Property

10.2.4.1 Filter string

```
SELECT * FROM CIM_InstModification
    WHERE SourceInstance ISA {class-name}
    AND ANY
    SourceInstance.{class-name}::{property-name}[*] = {value1}
    AND ANY
    SourceInstance.{class-name}::{property-name}[*] = {value2}
```

10.2.4.2 Semantic

The array property {property-name} in class {class-name} (or one of its subclasses) has been modified resulting in one of the entries in the array having a value of {value1} and another of the entries having a value of {value2}. Either {value1} or {value2} shall be a new value for an existing entry or is the value of a newly added entry.

10.2.5 Alert

10.2.5.1 Filter String

```
TSELECT * FROM CIM_AlertIndication WHERE OwningEntity='SNIA'  
      AND MessageID='{message-id}'
```

10.2.5.2 Semantic

An alert indication referencing the standard message with message ID {message-id}. Note that the message ID is a concatenation of the name of the appropriate SNIA registry and message number. For example, the {message-id} for the first message in the FC registry is 'FC1'.

11 SMI-S Roles

11.1 Introduction

As shown in Figure 22, the complete reference model shows the roles for the various entities of the management system. Any given host, network device or storage device may implement one or more of these roles as described later in this clause.

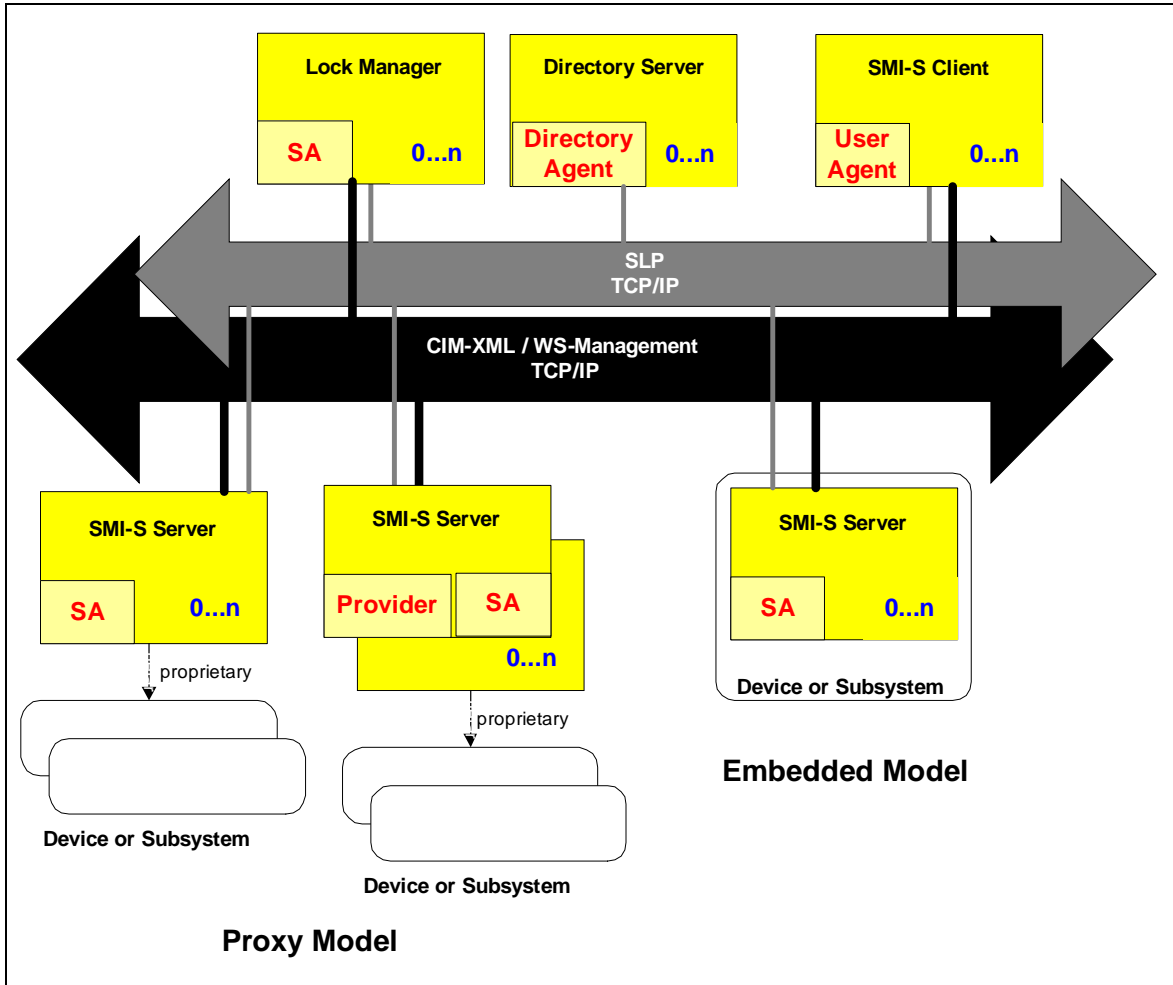


Figure 22 - SMI-S Roles

This profile presents a concise definition of each of these roles and the requirements on implementations of these roles in a management system. For each of these roles, specific functions are required to be implemented in one or more functional areas:

- SLP Discovery Functions – the required discovery capabilities that the role performs in the overall management system;
- Generic Operations – the management model operations that the role performs;
- Security – the security requirements that the role is expected to satisfy;
- Lock Management Operations – the locking operations that the role is expected to perform.

The detail of these responsibilities for each of the roles is described in this profile.

11.2 SMI-S Client

11.2.1 Overview

The SMI-S Client role in the overall management system is performed by software that is capable of performing management operations on the resources under management. This includes monitoring, configuration, and control of the operations on the resources. Typical clients include user interface consoles, complete management frameworks, and higher-level management applications and services such as policy based management systems.

There can be zero or more SMI-S clients in the overall management system. These clients can all coexist simultaneously and can perform independent or overlapping operations in the management system. It is outside the scope of this specification to specify client cooperation with other clients in any way. The semantics of the described management system is that the last successful client operation is valid and persists in the absence of any other client operations (last write wins).

It is expected that development kits for the management system will provide code for the required functions implemented in clients. Consoles, frameworks and management applications can then use this common code in order to comply with this specification. The specification of an API for this client code, and specific language bindings for applications is outside the scope of this specification, but is a candidate for follow-on work.

11.2.2 SLP Functions

The SMI-S Client role is required to implement SLP User Agent (UA) functionality as specified in 9.6, "User Agents (UA)". The Client discovers all SMI-S servers within its configured scope that are required for its operations by querying for service specific attributes that match the criteria for those operations.

11.2.3 Generic Operations

The SMI-S Client role shall implement client functionality as specified by the relevant WBEM protocol standard and should implement asynchronous notification functionality as specified by that standard.

11.2.4 Security Considerations

The SMI-S Client role shall implement security as specified in 13.2.2, "General Requirements for HTTP Implementations".

11.2.5 Lock Management Functions

Not defined in this standard.

11.3 Dedicated SMI-S Server

11.3.1 Overview

The intention of the SMI-S server role in a management system is to provide device management support in the absence of any other role. A simple management system could consist of just a SMI-S Client and a SMI-S Server and all management functions can be performed on the underlying resource. This means that a vendor can offer complete management for the resource by shipping a standalone client for the resource and not depend on any other management infrastructure. Although, at the same time, the SMI-S Server can participate in a more complex management environment through the use of the standard mechanisms described here.

- Embedded SMI-S Server – the SMI-S Server functions are incorporated into the resource directly and do not involve separate installation steps to become operational.
- Proxy SMI-S Server – the SMI-S Server is hosted on a system separate from the resource and communicates with the resource via either a standard or proprietary remote protocol. This typically involves

an installation operation for the SMI-S Server and configuration for, or independent discovery of, the desired resource.

In order to minimize the footprint on the resource or proxy hosts, the required functions of the SMI-S Server role have purposely been scaled back from those of a typical general purpose WBEM Server running on host with more significant resources. These required functions are described in 11.3.2 and 11.3.3.

11.3.2 SLP Functions

The SMI-S Server role is required to implement SLP Service Agent (SA) functionality as specified in 9.7, "Service Agents (SAs)". Optionally, it should implement Service Agent Server functionality or use an existing SA Server if one exists. The SMI-S server shall advertise service-specific attributes that allow the client to locate it based on its profile (see 9.5).

11.3.3 Generic Operations

11.3.3.1 General

The SMI-S Server role shall implement the server functionality as specified by the relevant WBEM Protocol standard.

11.3.3.2 Required Operations

SMI-S Servers shall support the following generic operations:

- GetInstance
- DeleteInstance
- ModifyInstance
- CreateInstance
- OpenEnumerateInstances
- OpenAssociators
- OpenReferences
- PullInstancesWithPath
- CloseEnumeration
- InvokeMethod
- InvokeStaticMethod

DEPRECATED

The following operations are deprecated. These operations shall still be supported in SMI-S 1.x versions, but will be removed in the future.

- EnumerateInstances
- EnumerateInstanceNames
- Associators
- AssociatorNames

- References
- ReferenceNames
- OpenEnumerateInstancePaths
- OpenAssociatorPaths
- OpenReferencePaths
- PullInstancePaths

DEPRECATED

11.3.3.3 Required Model Support

The SMI-S Server shall implement the Server Profile as detailed in *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 Server Profile*.

11.3.4 Security Considerations

The SMI-S Server role shall implement security as specified in 13 Security.

11.3.5 Lock Management Functions

Not defined in this standard.

11.4 General Purpose SMI-S Server

11.4.1 Overview

The General Purpose SMI-S Server role in an overall management system is intended to reduce the number of network connections needed by a Client to manage large numbers of resources. It is also envisioned as a convenient place to perform operations across multiple resources, further off-loading these from the Client as well.

In addition, the General Purpose SMI-S Server role can provide a hosting environment for the plug-in instrumentation of host-based resources and management proxies for resources with remote management protocols. These plug-ins are called providers and considered sub roles of the General Purpose SMI-S Server.

A General Purpose SMI-S Server is not required in a management system, but is expected to be deployed at least as a common infrastructure for host-based resources. In any large storage network, there may be several General Purpose SMI-S Servers (as many as one per host). Communication between General Purpose SMI-S Servers may be standardized in the future, but this capability is outside the scope of this specification. General Purpose SMI-S Servers may act as a point of aggregation for multiple SMI-S Profiles as described in 35 Server Profile using existing standard mechanisms as specified here.

As General Purpose SMI-S Servers are expected to be deployed on hosts with more resources and less footprint concerns than other managed resources, the required functions, specified in 11.4.2, 11.4.3, and 11.4.4, are more extensive that of an Dedicated SMI-S Server.

11.4.2 SLP Functions

The General Purpose SMI-S Server role is required to implement SLP Service Agent (SA) functionality as specified in 9.7, "Service Agents (SAs)". The General Purpose SMI-S Server shall advertise service specific attributes that allow the Client to locate it based on the profiles it supports, as defined in 11.4.3.1, "General".

11.4.3 Generic Operations

11.4.3.1 General

The General Purpose SMI-S Server role shall implement WBEM Server functionality as specified by the Generic Operations standard.

11.4.3.2 Required Operations

The General Purpose SMI-S Server is required to implement the minimum profile as specified in the Generic Operations standard. In addition, it shall implement the intrinsic methods needed to support the Profiles that it supports.

11.4.3.3 Required Model Support

The General Purpose SMI-S Server shall implement the Server Profile as detailed in *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 Server Profile*.

11.4.3.4 Security Considerations

The General Purpose SMI-S Server role shall implement security as specified in *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 Server Profile*.

11.4.4 Lock Management Functions

Not defined in this standard.

11.4.5 Provider Sub-role

11.4.5.1 Overview

A sub-role within a General Purpose SMI-S Server that can be used to provide management support for the resource, especially useful when the resource is host-based (i.e., HBA or Host Software) and the platform provides a WBEM Server as part of its operating system.

11.4.5.2 Required Model Support

The Provider shall implement the Provider Profile as detailed in the object model shown in *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 Server Profile*.

11.5 Directory Server

The Directory Server role is used to facilitate Discovery of instances of the various roles in a management system, but may also be used by management systems to store common configurations, user credentials and management policies. Functions outside of Discovery are outside the scope of this specification. The Directory Server role is optional for a compliant management system.

11.5.1 SLP Functions

The Directory Server role is required to implement SLP Directory Agent (DA) functionality as specified in 9.8, "Directory Agents (DAs)". The Directory registers all Agents and Object Managers within its configured scope and allows queries for their respective service specific attributes.

11.5.2 Generic Operations

There are no additional requirements for this role.

11.5.3 Security Considerations

There are no additional security requirements for this role.

11.5.4 Lock Management Functions

Not defined in this standard.

11.6 Combined Roles on a Single System

11.6.1 Overview

As mentioned previously, the various roles of the management system can be deployed in different combinations to different systems throughout the managed environment. In general, there are no restrictions on what roles can be deployed on any given system, but some examples are given to illustrate typical situations.

11.6.2 General Purpose SMI-S Server as a Profile Aggregator

11.6.2.1 SLP Functions

The General Purpose SMI-S Server role may implement SLP User Agent (UA) functionality as specified in 9.6, "User Agents (UA)". The General Purpose SMI-S Server discovers all Profiles within its configured scope that are aggregated by querying for service specific attributes that match the criteria for those aggregations.

11.6.2.2 Generic Operations

The General Purpose SMI-S Server role may implement CIM Client functionality as specified by Generic Operations standard and may implement CIM listener functionality as specified by the applicable WBEM Protocol standard. A General Purpose SMI-S Server may reflect instances and classes from the aggregated Profiles (perhaps by delegating operations to the Dedicated SMI-S Servers), but is not required to do so. The Profile's Model instances should be reflected in the advertised default namespace of the General Purpose SMI-S Server. The hierarchy of General Purpose SMI-S Servers and Dedicated SMI-S Servers in a multi-level system needs to be reflected in the model such that it can be administrated.

11.6.2.3 Security Considerations

There are no requirements for security for this role.

11.6.2.4 Lock Manager Functions

There are no requirements for locking in this release of the specification.

12 Installation and Upgrade

12.1 Introduction

The interoperability of the management communications in a storage network gives customers a choice in vendors of their management solutions, but it also can introduce ease-of-use problems when these different vendors each supply different components. In order to supply a complete management solution, many management vendors provide not only WBEM Clients, Providers and other Management Interfaces, but also software components that provide other pieces of the management infrastructure (e.g., Directory Services, WBEM Services, Database Management). Problems are possible when multiple vendors install or remove these components in the same configuration and conflicts can arise. One of the goals of creating management interoperability is to reduce the time and expense end-users apply to the management of their SANs. Thus, SAN management should be easy to install, easy to upgrade, and easy to reconfigure. Mature management products using SMI-S technology should experience seamless and almost completely automated installation, upgrade, and reconfiguration.

This clause deals with issues in installation, upgrade and uninstallation of products using SMI-S technology, and recommends some steps that vendors should take to minimize the problems, leading to better customer satisfaction with the overall management solution.

12.2 Role of the Administrator

Ultimately, a vendor's installation software cannot make perfect decisions when the conflicts referenced in 12.1 arise, since there may be valid reasons why a customer has deployed software of similar function from multiple vendors. In the situation where two software components are both installed that perform the same shared function, and only one can reasonably operate without conflicts, the administrator must be able to resolve these conflicts and remove or disable the redundant component(s).

Installation software should, however, make a best effort to detect any conflicts and notify the administrator of possible conflicts during its installation and initialization. A vendor's installation software should allow the administrator to install and uninstall the various infrastructure components on an individual basis should such a conflict arise. The implications of this are that vendors are motivated to support interoperation with other vendor's components. The advantage to the vendor is that a customer is more likely to install a component that can demonstrate the most interoperability with other components.

12.3 Goals

12.3.1 Non-Disruptive Installation and De-installation

WBEM Clients & Services, Providers, and Directory Services may be capable of being installed and de-installed without disrupting the operation of other constituents in a SMI-S management environment. As SANs are often deployed in mission critical environments the up-time of the solution is critical and thus, the uptime of the management backbone as a key component of the solution is equally critical. Additionally, the installation and de-installation of SMI-S interface constituents should not compromise the availability of mission critical applications.

12.3.2 Plug-and-Play

The ultimate goal of management interoperability is zero administration of the management system itself. A customer should be able to install new storage hardware and software and have the new component become part of the management system automatically. Use of the Service Discovery process (see Clause 9 Service Discovery), the discovery-related aspects of the SMI-S Role definitions (see Clause 11 SMI-S Roles), and the Server Profile (see *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 Server Profile*) are intended to assist in achieving this goal.

During the reconfiguration of the management system, the schema that Clients see should remain consistent (Schema forward compatibility is ensured via CIM standard).

12.4 Server Deployment

12.4.1 General

Manufacturers of storage hardware and software typically install their product and the accompanying management support at the same time. Conflicts are possible between Agents if multiple vendors attempt to install support for the same device. Also, when a device vendor needs to upgrade an Agent or Provider for a device, the installation software needs to determine all of the locations of the previous installations to insure there are not duplicate management paths to the device and thus, insure reliable on-going operation of the device. SMI-S agent implementations must first determine the environment they will run in and how they will be installed into that environment. These environments include:

- Controlled environment (12.4.2)
- Multiple WBEM Server environment (12.4.3)
- Shared WBEM Server environment (12.4.4)

For each of these issue areas, this clause provides requirements to authors of SMI-S agents and CIM-based management software. These practices are designed to maximize interoperability.

12.4.2 Controlled Environment

A Controlled environment is either embedded in the system being managed or a dedicated management processor that limits the software a user can install on it. Agents in controlled environments shall be exempt from the requirements in 12.4.3 and 12.4.4.

12.4.3 Multiple WBEM Server systems

A system supporting multiple CIM agents may require multiple WBEM Servers. Because the SMI-S agent can't control when multiple WBEM Servers are required, all SMI-S agents other than controlled environments shall implement the Multiple WBEM Server requirements.

12.4.3.1 Determine Multiple WBEM Servers

Installation software for devices shall be able to locate existing WBEM Servers that may control the device in order to offer an administrator a choice in management constituents for the device. In addition, the installation software should locate existing Agents and Providers that provide device support in order to reliably upgrade that support. For these reasons, an installation software program may act as a SMI-S Client during installation. This will allow it to employ the Service Discovery (see Clause 9 Service Discovery) to locate the appropriate functions, and to make the automated decisions that eliminate the need for an administrator to manually configure or adjust certain aspects of the management system.

12.4.3.2 Ports

SMI-S uses TCP/IP, HTTP/HTTPS, and CIM-XML or WS-Management protocols. These protocols require the use of TCP/IP ports. SMI-S defines the way a client discovers the server ports in 9 Service Discovery. Any SMI-S agent (WBEM Server) that may be installed in an environment with other agents shall support the use of alternate port addresses. The agent shall support user configured port addresses.

12.4.3.3 SLP

SMI-S requires the use of SLP for agent discovery (see Clause 9 Service Discovery). The SLP standard requires the use of a "well known port" that may not be shared. Therefore, a computer system can only have one instance of a SLP service agent running on a system. All SMI agents on the system shall register with the common SLP service agent or provide user documentation that allows a user to manually register the agent and its profiles.

12.4.3.4 Directories

Some environments require multiple copies of the same WBEM Server to be installed. This may be done to solve version compatibility issues. SMI agents shall be coded to allow user settable directory names to be used. Installation programs for SMI agents should find all instances of compatible WBEM Servers and allow the user to select the WBEM Server installed into. The installation shall then install the agent in directories relative to that WBEM Server.

12.4.3.5 Miscellaneous

Conflicts are possible between Agents if multiple vendors attempt to install support for the same device. Also, when a device vendor needs to upgrade an Agent or Provider for a device, the installation software shall determine all of the locations of the previous installations to insure there is not duplicate management paths to the device and thus, insure reliable on-going operation of the device.

12.4.3.6 Tools

Utilities needed to manage the WBEM Server (e.g., users, configuration) shall be able to find the WBEM Server and allow the user to select a WBEM Server if more than one is found.

12.4.4 Shared WBEM Server

A shared WBEM Server environment is when two or more unrelated providers share a single WBEM Server.

12.4.4.1 Namespaces

In the case of shared WBEM Servers, namespaces help isolate implementations and reduce provider interaction. The device model should be implemented in a vendor specific namespace. A single vendor may choose to put multiple implementations in its own namespace. Vendor namespace names should be chosen to reduce any chance of conflict. The namespace name should include the vendor's company name or stock symbol.

12.4.4.2 Trivial sub classes

"CIM" classes should not be implemented directly. They should be subclassed using a name prefix unique to their company. This sub classing prevents interaction between provides. Instances in the "interop" namespace shall be subclassed.

12.4.4.3 "interop" namespace

The profile registration profile shall be implemented in a namespace named "interop". The profile contains two parts. First part is a model of the WBEM Server. This section shall be implemented by the software package that installs the WBEM Server. All other implementations shall extend the profile registration profile with instances that define the profile they support.

Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3, 36.3.4 The SMI-S Registered Profile shows what device support is already installed, and installation software should consult this schema before installing new software. If the installation software is changing the device support from one configuration to another, the installation software needs to uninstall or disable the previous software support elements.

12.4.4.4 SLP support

Tenant providers shall extend the "Profile Registration Profile" and shall extend the SLP registered profiles as required in the SMI-S discover clause.

12.4.4.5 Version/Change control

It is the responsibility of the SMI agent installation to protect the WBEM Server. The installation process shall determine if a compatible WBEM Server is installed before becoming a tenant.

12.4.4.6 Base Server Profile

Some profiles can extend the “Base Server Profile”. New providers should look for a “Base Server Profile” to extend before installing its own.

12.4.5 Uninstallation

During the uninstallation of a device, the installation/uninstallation software (if available) should automatically detect existing management support software for the device in order to shut down and remove it in a consistent manner. This detection process need to be cognizant that SMI-S Clients may be actively using the device and that the device may need to be disabled for new management operations and administrated through an orderly shutdown procedure prior to uninstallation. The implementation of such procedures and any order dependency is outside the scope of this specification, but may need to be considered by implementers.

12.4.6 Update

During the update of device support software, installation software should automatically detect any existing device support software in order to successfully complete the upgrade. This device support may exist on multiple hosts, but that situation is not specified in this version. If the update includes installing a new provider, the installation software needs to use the provider installation/upgrade method that is supported by the existing Object Manager. When a software update involves a major schema version upgrade (e.g., 2.x to 3.x), the installation software needs to be cognizant of the effect of the schema upgrade on existing clients. For example, it may choose to simultaneously support both versions for some period of time.

12.4.7 Reconfiguration

When device support update requires an update of an agent or provider, the device support installation software should configure the new provider with the same subscriptions that exist in the old agent or provider before removing it, unless those subscriptions are specifically defined as being periodically cleaned up. This can be done via the instances of the subscriptions in the agent or object manager that currently exist.

12.5 WBEM Service Support & Related Functions

12.5.1 Installation

Customers are increasingly sensitive to the size of the memory footprint for management software. The goal is to minimize the impact on hosts that are not dedicated to running management software by making appropriate choices during installation and giving the administrator control over these issues.

It is recommended that vendors take advantage of an existing Object Manager where one exists, by installing a provider that communicated with that Object Manager for device support. Additional support for such “multi-tenant” Object Managers will be included in a future version of this document.

If an object manager does not exist, or the device support does not work with the existing object manager (e.g., due to interface requirements) it is recommended that the vendor supply a Agent that is lightweight for device support. Another option is to offer to install an Object Manager that the vendor does have provider support for, allowing other vendors to further leverage that installation.

Providers that use an in-band connection to devices have an issue where zoning may alter the management path to the device from a provider or agent. In this case, the device support may need to be installed on multiple hosts in the network and the vendor needs to provide some way to coordinate which provider or agent is responsible for a particular device.

Vendors should install their providers in a unique namespace for isolation and qualification reasons. The installer should employ the Service Discovery process (see Clause 9 Service Discovery), and/or the Server Profile (see *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35*

Server Profile) to discover the existing namespaces and insure that the one created for the new device is truly unique.

12.5.2 Multiple WBEM Servers on a Single Server System

At installation and setup, a user interface should be provided by the WBEM Server installation utility that allows an administrator to manually set the TCP port number in a persistent fashion.

To support discovery, the SLP Service Agent (see 9.7, "Service Agents (SAs)") associated with a newly-installed WBEM Server should register its TCP port number along with all the other necessary discovery information with the Discovery Service. This requirement applies to both automated port selection as well as manually configured installations. Clients, working through their SLP User Agent, described in 9.6, "User Agents (UA)", then use this information to establish contact with the WBEM Server.

12.5.3 Uninstallation/Upgrade

An Object Manager may be upgraded without needing to change the Providers that it supports. Depending on the Object Manager, the Providers may have to be reinstalled and reconfigured following such an upgrade. In this case, an administrator may need to re-run the device support installation software and such software should be able to restore the previous configuration.

12.5.4 Reconfiguration

Device Support Reconfiguration (see 12.3.2, "Plug-and-Play") identifies issues that may also be applicable to Object Managers.

12.5.5 Failure

Temporary failure of an object manager (for example, a host being powered off) can result in bad installation decisions for installation software. In this case, it is advisable that the installation software provide for manual input of the characteristics of additional components of the management system that the installation process needs to consider.

12.6 Client

12.6.1 Uninstallation

When Client software is removed, the uninstallation software should ensure that all client-defined information (settings, policies etc.), and any subscriptions for that client that exist in any agent or object manager, are also removed.

12.6.2 Reconfiguration

Client software can include a Listener that is configured to listen on a specific port. When this port is reconfigured, the client should redirect any Indication Handlers in existing agent and object managers as a result.

12.7 Directory Service

12.7.1 Installation

The installation of more than one Directory Agent—addressed in 9.6, "User Agents (UA)"—or Service Agent Server—addressed in 9.7, "Service Agents (SAs)"—providing a Directory Service in a management system does not impose a significant burden for management clients and adds to the overall availability. Vendors should recommend to administrators of their products that one or more SA Servers or Directory Agents should be deployed in the management system. This may also be done for network or system management reasons.

12.7.2 Uninstallation/Failure

SLP Clients are defined to handle failure and uninstallation of DAs as per the specification (see Clause 9 Service Discovery).

12.8 Issues with Discovery Mechanisms

Experience with existing SMI-S installations has indicated that some sites have policies that can impact the Service Discovery process (see Clause 9 Service Discovery). This subject will be addressed in greater detail in a future revision of this document, but two specific items of guidance are given here, as follows:

- a) Where the site policy has caused multicast to be disabled, the DHCP option for SLP defined in IETF RFC 2610 is recommended as an alternate method of locating Service Agent Servers or Directory Agents. Also note that the shipping configuration of many network routers has multicast disabled.
- b) Where the site policy has caused support for SLP itself to be disabled, an out of band method of providing a list of IP addresses for WBEM Servers is recommended, after which the Server Profile (see *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 Server Profile*) should be used to obtain the information about Registered Profiles usually retrieved via SLP.

IMPLEMENTED

13 Security

13.1 Objectives

Security in the context of SMI-S refers to the protective measures employed in the management of storage. The specific objectives to be addressed by security include:

- 1) Provide a mechanism that assures that the communications between a SMI-S client and server cannot be read or modified by a third party (confidentiality and integrity).
- 2) Provide a mechanism that allows SMI-S clients and servers to provide an assurance of their identity (authentication).
- 3) Provide a mechanism that allows control of the actions a SMI-S client is permitted to perform on a SMI-S server (authorization).
- 4) Provide a mechanism for records to be generated for actions performed by a SMI-S client on a SMI-S server (auditing).
- 5) Provide a mechanism that allows SMI-S clients to discover the SMI-S constituents in a storage network environment so that they may communicate with these constituents using CIM Operations over HTTP protocol.

13.2 Requirements

13.2.1 Overview

Security requirements can be divided into five major categories:

- 1) Authentication - verifying the identity of an entity (client or server)
- 2) Authorization - deciding if an entity is allowed to perform a given operation
- 3) Confidentiality - restricting information to only those intended recipients
- 4) Integrity - guaranteeing that information, passed between entities, has not been modified
- 5) Non-repudiation - the ability to prove an action or event has taken place, so that this event or action cannot be denied or disavowed later.

SMI-S security primarily addresses authentication, confidentiality of communications, and authorization to a lesser degree. Integrity has been left for future work, and non-repudiation is not currently identified as a need for SMI-S.

Security concerns occur in three areas of an SMI-S implementation:

- 1) First, an SMI-S Server may also be a client of other services (sometimes conceptualized as a device). Those services, or devices, may require a login before discovery or operations are allowed to be performed. The information needed to perform this login is generically referred to as "credentials" (or in the case of devices as "device credentials"). An SMI-S server or provider needs to obtain these credentials in order to talk to the service, and they should be provided confidentially.
- 2) Second, an SMI-S Server may need to authenticate an SMI-S Client. Not all Clients may be allowed to query the object model, and not all Clients may be allowed to perform operations on objects in the model. The SMI-S Server is responsible for the process of authenticating credentials received from an SMI-S Client. Successful authentication establishes a trust relationship, which is represented on

the SMI-S Server by an authenticated Identity. Authenticating the client is the first step in determining what that Client is allowed to do.

- 3) Thirdly, should implementers of an SMI-S Server be unaware of secure development practices, attackers may be able to exploit resulting flaws in implementations.

13.2.2 General Requirements for HTTP Implementations

The security requirements for HTTP implementations apply to both SMI-S servers and clients. An SMI-S client shall comply with all security requirements for HTTP that are applicable to clients. The following are general requirements for the support of security when using HTTP.

- 1) SMI-S Servers and Clients shall conform to *DMTF DSP0200 CIM Operations over HTTP*.
- 2) HTTP Basic Authentication shall be implemented. HTTP Digest Authentication should be implemented. See 13.3.2.1 "User Authentication".
- 3) To minimize compromising user identities, and credentials such as passwords, implementations should use HTTP Basic Authentication ONLY in conjunction with TLS.
- 4) Where TLS is not used, implementers should utilize HTTP Digest Authentication. See 13.3.2.1 "User Authentication".
- 5) A user identity and credential used with one type of HTTP Authentication (i.e., Basic or Digest) shall not ever be subsequently used with the other type of HTTP Authentication. To avoid compromising the integrity of a stronger scheme, established good security practices avoid the reuse of identity & credential information across schemes of different strengths. See 13.3.2.1 "User Authentication".
- 6) TLS as specified in *SNIA TLS Specification for Storage Systems* shall be implemented and should be used.
- 7) Clients that fail to contact an SMI-S server via HTTP over TLS on TCP port 5989 should retry with HTTP on TCP port 5988 if their security policy allows it.
- 8) In order for Clients and Servers to communicate, they need to be using a consistent approach to security. It is possible for properly configured Clients and Servers to fail to communicate if one is relying upon port 5989 and the other on port 5988.
- 9) Servers can accelerate discovery that a secure channel is needed by responding to HTTP contacts on TCP port 5988 with an HTTP REDIRECT to the appropriate HTTPS: URL (HTTP over TLS on TCP port 5989) to avoid the need for clients to timeout the HTTP contact attempt. Clients should honor such redirects in this situation.

13.3 Description of SMI-S Security

SMI-S security is primarily focused on securing the underlying network transport, authenticating users, and securely interacting with IT infrastructure.

13.3.1 Transport Security

For most SMI-S implementations, the Hypertext Transfer Protocol (HTTP) is the underlying communications protocol used to transfer SMI-S messages, but it is possible that other transports like Web Services for System Management (WS-Management) may be used. A major element of SMI-S s

CIM over HTTP is the mandatory transport mechanism for this version of SMI-S and the specific requirements are security is focused on securing these underlying transports.

derived from DMTF DSP0200, (Specification for CIM Operations over HTTP), which describes the requirements for CIM clients and servers. It is important to note that HTTP by itself offers no confidentiality or integrity protections.

SMI-S also includes a mechanism to secure HTTP communications such that data sent between the clients and servers are encrypted before being sent out over the network. This security is achieved by transmitting HTTP over SSL/TLS (also known as HTTPS); the URL of a secure connection will begin with https:// instead of http://. It is also important to note that an SMI-S Client communicates with an SMI-S server via HTTPS on TCP port 5989 (TCP port 5988 is used for HTTP).

When TLS is used to secure HTTP, the client and server typically perform some form of entity authentication. However, the specific nature of this entity authentication is dependent on the cipher suite negotiated; a cipher suite specifies the encryption algorithm and digest algorithm to use on a SSL/TLS connection. A very common scenario involves the use of server-side certificates, which the client trusts, as the basis for unidirectional, entity authentication. It is possible that no authentication will occur (e.g., anonymous authentication) or on the other extreme, mutual authentication involving both client-side and server-side certificates may be required.

STABLE

The specific requirements and options associated with the implementation of TLS within conformant SMI-S systems are specified in *SNIA TLS Specification for Storage Systems*. The use of these TLS features is strongly encouraged.

13.3.2 Authentication

At a basic level, authentication is the process used to identify a user (or entity) through the verification of supplied information (i.e., verify a declared identity). This information is often a secret (e.g., a password), but it may also be accomplished by possessing something (e.g., a smart card) or be something that you are (e.g., biometrics); combining multiple forms (or factors) of authentication credentials is known as multi-factor authentication. Increasingly, strong (multi-factor) authentication is required for privileged users or any remote access (including vendor access). It is also important to note that some of these credentials are static (i.e., indefinite use period) while others have expiration periods or may be one-time-use.

Within SMI-S, the dominant form of authentication is for users, but entity authentication does occur. In addition, the SMI-S Servers frequently employ local authentication, but external authentication is an option.

STABLE

13.3.2.1 User Authentication

SMI-S Clients are responsible for initiating user authentication for each SMI-S Server that is accessed by a user. HTTP Basic Authentication shall be implemented and HTTP Digest Authentication should be implemented; HTTP Digest Authentication is a required contingency when authentication credentials have to be secured, but appropriate SSL or TLS protections cannot be negotiated. For both forms of

HTTP authentication, the SMI-S Server functions as the authenticator and it receives the user credentials from the HTTP authentication operations.

Established good security practices avoid the reuse of identity & credential information across schemes of different strengths. Thus, a SMI-S user identity and credential used with one type of HTTP Authentication (i.e., Basic or Digest) shall not ever be subsequently used with the other type of HTTP Authentication.

Section 4.4 of DSP0200 defines additional requirements for HTTP authentication, above those found in IETF RFC 2616 or IETF RFC 2617. HTTP authentication generally starts with an HTTP client request, such as "GET Request-URI" (where Request-URI is the resource requested). If the client request does not include an "Authorization" header line and authentication is required, the server responds with a "401 unauthorized" status code, and a "WWW-Authenticate" header line. The HTTP client shall then respond with the appropriate "Authorization" header line in a subsequent request. The format of the "WWW-Authenticate" and "Authorization" header lines varies depending on the type of authentication required: basic authentication or digest authentication. If the authentication is successful, the HTTP server will respond with a status code of "200 OK".

Basic authentication involves sending the user name and password in the clear, and should only be used on a secure network, or in conjunction with a mechanism that ensures confidentiality, such as TLS. Digest authentication sends a secure digest of the user name and password (and other information including a nonce value), so that the password is not revealed. "401Unauthorized" responses should not include a choice of authentication.

Client authentication to the SMI-S Server is based on an authentication service (local and/or external). Differing authentication schemes may be supported, including host-based authentication, Kerberos, PKI, or other; the authentication service is out of scope of this specification.

STABLE

13.3.2.2 Entity Authentication

Entity authentication is the process by which an agent in a distributed system gains confidence in the identity of a communication partner. More often than not, the entity authentication process is coupled with the distribution of a "session key" which the partners can later use for message confidentiality, integrity, or whatever else.

Within SMI-S, entity authentication is typically performed whenever SSL/TLS is used and it is accomplished using digital certificates. An SMI-S server may also use a form of entity authentication for certain types of third-party authentications services. For example, RADIUS employs a shared secret to protect certain user credentials.

13.3.3 Service Discovery

Service discovery protocols are network protocols which allow automatic detection of devices and services offered by these devices on a computer network. Within the context of SMI-S (see Clause 9 Service Discovery), service discovery refers to the discovery of dedicated SMI-S servers, general purpose SMI-S servers, and directory servers as well as the functions they offer in an SMI-S managed environment. This release of SMI-S uses the Service Location Protocol Version 2 (SLPv2), as defined by IETF RFC 2608, for its basic discovery mechanism.

SLP is a packet-oriented protocol. Most packets are transmitted using UDP, but TCP can also be used for the transmission of longer packets. Because of the potential unreliability of UDP, SLP repeats all multicasts several times in increasing intervals until an answer has been received. All devices are required to listen on port 427 for UDP packets, SAs and DAs should also listen for TCP on the same port. Multicasting is used extensively by SLP, especially by devices that join a network and need to find other devices.

The operation of SLP differs considerably, depending on whether a Directory Agent (DA) is in the network or not. When a client first joins a network, it multicasts a query for DAs on the network. If no DA answers, the client will assume that it is in a network without DAs. It is also possible to add DAs later, as they multicast a "heartbeat" packet in a predefined interval that will be received by all other devices. When a SA discovers a DA, it is required to register all services at the DA. When a service disappears the SA should notify the DA and un-register it.

The SLPv2 security model assumes that service information is public, and therefore does not require confidentiality. SLPv2 provides for authentication of service URLs and service attributes, thus providing integrity assurances for service URLs and attributes included in SLP messages. For SMI-S environments that require security in conjunction with the use of SLPv2, the major threat mitigation strategies (see RFC 3723) are not necessary as long as the SLP messages are not fully trusted and SSL/TLS with server certificates is used. Additional security guidance is provided in 9 Service Discovery.

IMPLEMENTED

Entity Authentication

Annex A (informative) SMI-S Information Model

This standard is based on DMTF's CIM schema, Version 2.51. 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 B (normative) Compliance with the SNIA SMI Specification

B.1 Compliance Statement

The declaration of SMI-S compliance of a given CIM Instance within a WBEM Server also declares that any CIM Instance associated, directly or indirectly, to the first CIM Instance will also be SMI-S compliant if SMI-S itself declares compliance rules for either CIM Instance or instances of their superclasses. The declaration of SMI-S compliance also declares that the implementation shall also conform to the SMI-S architecture as defined in *Storage Management Technical Specification, Part 2 Common Architecture*.

B.2 How Compliance of the Architecture is Declared

An agent indicates which version of SMI-S it conforms to using “the SMI-S registered profile” as defined in the Profile Registration Profile (see *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 36.3.4 "The SMI-S Registered Profile"*). The agent shall conform to all the provisions of the versions of *Storage Management Technical Specification, Part 2 Common Architecture* where it instantiates an instance of RegisteredProfile with a matching RegisteredVersion value.

B.3 How Compliance of the Model Is Declared

- The declaration of SMI-S compliance is made through the use of the Server Profile and the declaration of supported profiles.
- Direct association between CIM Instances is made through instance of a CIM Association.
- Indirect association between CIM Instance is made through more than one CIM Association.
- SMI-S Compliance is assessed against CIM Instances that are directly or indirectly associated to the CIM Instance declared as part of the declaration of supported registered profiles. These CIM Instances comprise the compliance test set.
- All CIM Instances / CIM Classes included in the compliance test set for whom compliance rules are defined in SMI-S or for superclasses thereof shall be themselves be compliant to the rules defined in SMI-S.
- Compliance tests on a superclass of a given CIM Instance are limited to the attributes and behaviors defined for the superclass.

B.4 The Server Profile and Compliance

B.4.1 General Guidelines

Compliance is declared by the implementation of the Server Profile. All profiles require the Server Profile. The Server Profile defines the means by which a SMI-S Client determines the profiles supported and the ComputerSystems associated. (see *Storage Management Technical Specification, Part 3 Common Profiles, 1.8.0 Rev 3 35 "Server Profile"* for more details.)

B.4.2 Example

A CIM Agent for Vendor X declares compliance to the Array Profile and the Pool Manipulation Capabilities, and Setting profile through the Server Profile. Once the association (via the ElementConformsToProfile association) is made to from the Array Profile declaration to the ComputerSystem that realizes the Array Profile, then compliance tests begin testing compliance. Vendor X decided to extend the StorageVolume class with additional properties. StorageVolume is associated to the ComputerSystem via SystemDevice association. ComputerSystem, StorageVolume, and SystemDevice are defined in SMI-S as required CIM elements (see *Storage Management Technical Specification, Part 4 Block Devices, 1.8.0 Rev 3 Table 2, "CIM Elements for Array"*).

In implementing FCPort, Vendor X decided to not provide ElementName but did provide the rest of the required properties. Vendor X decided to not use WWN and instead used a vendor specific value for the PermanentAddress (see 7 Correlatable and Durable Names) Additionally, Vendor X added FRUStatus to their subclass of FCPort. Vendor X also decided to model the back-end fibre channel, but not use an SMI-S model to do so. These back-end FCPorts are associated to the ComputerSystem via the ConsumedSystemDevice association, a subclass of SystemDevice without properties overridden. These back-end fibre channel ports were modeled using a Vendor X specific class, BackendFCPorts, that is not derived from FCPort. This BackendFCPorts were associated to the ComputerSystem with the ConsumedSystemDevice.PartComponent role.

The compliance test includes FCPort because compliance declaration identified a particular ComputerSystem the entry point into compliant CIM instantiation of the Array Profile. the compliance test includes FCPorts as part of the test set because the SystemDevice association, also defined as part of the profile, includes the FCPort realized in that implementation. The compliance test also includes BackendFCPorts because the ConsumedSystemDevice association to the ComputerSystem for these instances is a SystemDevice association.

The compliance test locates the StorageConfigurationService, StoragePools including a Primordial StoragePool, and StorageCapabilities associated to the ComputerSystem. Vendor X's implementation supports the creation of a StoragePool. The test attempts to create a StoragePool given one of the sizes reported by the Primordial StoragePool.getSupportedSizes() method using the Primordial StoragePool reference and a StorageSetting generated from one of the StorageCapabilities.

The compliance test for Vendor X's Array Profile implementation fails because:

- FCPort.PermanentName property has a noncompliance value. Specifically, the FCPort.PermanentAddress is required to be WWN, 16 unseparated uppercase hex digits;
- ElementName property was not provided (i.e., was null);
- the SystemDevice associations contained references to BackendFCPort in the PartComponent property. CIM defined that the PartComponent is a LogicalDevice. Since BackendFCPort is not a LogicalDevice, then the test failed;
- The "Size not supported" return code was returned from CreateOrModifyStoragePool even though one of the supported sizes was used verbatim.

The compliance test for Vendor X's Array Profile implementation did not fail because:

- StorageVolume was extended;
- SystemDevice was extended.

B.5 Backward Compatibility

B.5.1 Goals

Backward compatibility between versions of SMI-S profiles is a requirement with very few exceptions. The goals of backwards compatibility include:

- a) New profile implementations that are deployed in a customer environment work with existing SMI-S Clients. This includes:
 - 1) SMI-S operations, including CTP, continue to work against the new profile implementation;
 - 2) SMI-S Clients can support a given profile version and above (later minor version numbers);
- b) No guarantee of backwards compatibility is implied between major version numbers (i.e., 1.x to 2.x);

- c) If a profile in a newer version of SMI-S cannot maintain backward compatibility, it shall be renamed (and the old profile deprecated). Otherwise the client may assume that the newer profile is backwards compatible and that all operations in the earlier version will continue to work in this newer version.
- d) It shall be possible for SMI-S provider and client implementations to support older versions of an incompatible profile.
- e) Content marked experimental is not standard in this version of the specification. Future versions of the specification may not be backwards compatible to content marked experimental in this version. Content marked experimental in this version of the specification may be removed in a future version.

B.5.2 Overview

SMI-S backward compatibility is necessary to ensure that customer environments are minimally disrupted by newer implementations of SMI-S. Deployment of several concurrent implementations of multiple minor versions of SMI-S shall be possible in a customer environment. Compatibility is required from both the Client side and from the provider side. Compatibility also has aspects both in the specification of newer functionality via SMI-S and in the implementation of both providers and clients.

Figure B.1 shows the interaction between a Client coded to an older minor version of SMI-S (M.m) acting against a later minor version (M.n) provider implementation

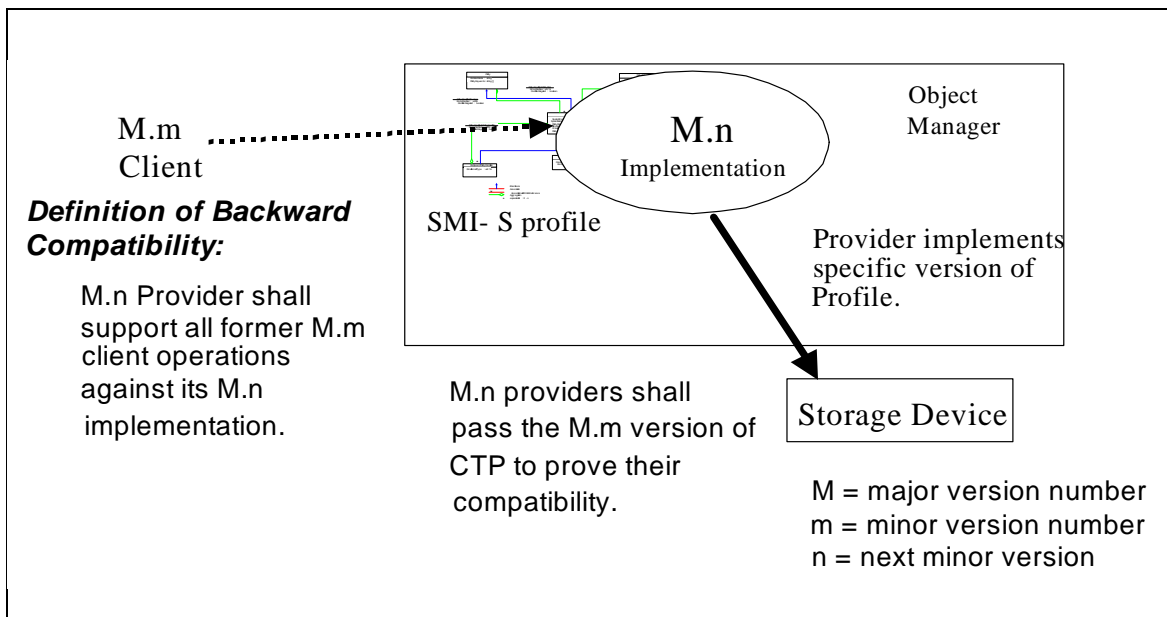


Figure B.1 - Provider Migration

As shown in Figure B.1, the newer implementation shall support all of the old operations from the previous minor version of SMI-S in order to maintain compatibility. The Client will not be able to take advantage of any newer features that have been added in the later version of the specification, but will still be able to accomplish all of the functions it was coded for in the previous version. This allows minimum disruption to the customer environment.

Clients shall be written to take advantage of the functionality of implementations that are currently shipping and that are or will soon be deployed in customer environments. This client functionality needs to be careful in how it makes use of each SMI-S version's new features. Any client code that uses a specific version's features shall also include a version check against the profile version in the

RegisteredProfile instance for that functionality. This version check shall verify that the functionality is at a specific minor version and above (up to the next major release). If a client were only to check for a specific version, it would not be able to use newer implementations of that functionality. A client will, over time, contain multiple such code blocks as newer versions are supported. Each piece of code will be written to the functionality introduced in a specific version and continue to work against that functionality in later minor releases.

B.5.3 Requirements

In order to maintain backwards compatibility with older minor versions of the specification, profile authors have followed specific rules in developing the specification. The requirements that were followed in profile versioning and shall be followed by subsequent implementations include:

- **Support for required classes:** A newer minor version of an SMI-S profile shall support all required classes of the previous minor version of the profile and shall continue to require them.
- **Support for conditional classes:** A newer minor version of an SMI-S profile shall support all conditional classes of the previous minor version of the profile and shall continue to require them as specified in the conditions of the previous minor version. But the newer minor version may add other conditions under which the class will be required. In addition, conditional classes in a previous minor version may be promoted to required in a newer minor version.
- **Support for optional classes:** A newer minor version of an SMI-S profile may promote a class to Conditional or Mandatory any class that was optional in the previous minor version.
- **Deprecation of classes:** A newer minor version of an SMI-S profile may deprecate or include deprecated (via the CIM schema) classes introduced in previous minor version(s), but shall continue to require their implementation.
- **Support for required properties:** A newer minor version of an SMI-S profile shall support all required properties of classes in the previous minor version(s) of the profile and shall continue to require them.
- **Support for conditional properties:** A newer minor version of an SMI-S profile shall support all conditional properties of classes in the previous minor version(s) of the profile and shall continue to require them as specified by the conditions of the previous minor version. But the newer minor version may add other conditions under which the property will be required. In addition, conditional properties in a previous minor version may be promoted to required in a newer minor version.
- **Support for optional classes:** A newer minor version of an SMI-S profile may promote a class to Conditional or Mandatory any class that was optional in the previous minor version.
- **Deprecation of properties:** A newer minor version of an SMI-S profile may deprecate or include deprecated (via the CIM schema) properties of classes introduced in previous minor version(s), but shall continue to require their implementation.
- **Support for component profiles:** A newer minor version of an SMI-S profile shall support the functionality of all component profiles of the previous minor version(s) of the profile and shall continue to require them if they were required in the previous version. A newer minor version of an SMI-S profile may require a component profile that was optional or conditional in the previous minor version, but shall not make optional or conditional a component profile that was required in a previous minor version. If a newer minor version of an SMI-S profile does not have component profiles by the same name as previous minor version(s), it shall still require implementation of the Registered (Sub)Profile with the previous version information such that the client will be able to find and use the subsumed functionality.
- A newer minor version of an SMI-S profile shall support all conditional component profiles of the previous minor version of the profile and shall continue to require them as specified in the conditions of the previous minor version. But the newer minor version may add other conditions under which the component profile will be required.

- **Profile renaming:** A newer minor version of an SMI-S profile that cannot remain backwards compatible shall either become a major revision of the profile or shall be renamed to a different profile name such that a client will not find newer, incompatible, versions of that functionality.

B.5.4 Implementation Considerations

Even in the case of a newer minor version of an SMI-S profile that was unable to retain backward compatibility, an implementation may support clients with a separate implementation of the previous minor version's functionality. Implementations shall not implement these earlier versions in such a way that a client of the previous minor version would become confused or break when accessing this functionality. This may happen if the previous version's functionality is implemented in the same namespace as the later version, but a careful evaluation needs to be done by the implementer to determine this.

B.6 Rules for Combining (Autonomous) Profiles

B.6.1 General

SMI-S specifies the behavior of (autonomous) profiles. The rules for compliance and backward compatibility are defined in the context of a profile (an Autonomous Profile). This subclause defines the rules that shall be applied when a device (or program) wishes to support the behavior of multiple (autonomous) profiles.

The guiding principles in such support are:

- Maintain Compliance (see B.1 through B.4)
 - Combining (autonomous) profiles shall not break compliance rules for any of the combined individual profiles.
- Maintain Backward Compatibility (see B.5)
 - Combining (autonomous) profiles shall not break backward compatibility for any of the combined individual profiles.

B.6.2 Backward Compatibility Rules for combining profiles

The backward compatibility rules apply to combined profiles in that combined profile implementations that are deployed in a customer environment shall work with SMI-S clients of any one of the profiles that were combined:

- **Support for required classes:** A combination of SMI-S profiles shall support all required classes of the individual profiles that have been combined and shall continue to require them. If a class is required in one individual profile, it shall be required in the combination profile.
- **Support for conditional classes:** A combination of SMI-S profiles shall support all conditional classes of the individual profiles that have been combined and shall continue to require them as specified in the conditions of individual profiles that have been combined. If a class is conditional in one or more of the individual profiles (and not required in any other individual profile) then it shall be conditional in the combination profile. If a class is conditional in multiple individual profiles, but with different conditions, then all conditions shall yield the existence of the class.
- **Deprecation of classes:** A combination of SMI-S profiles shall include any deprecated (via the CIM schema) classes introduced by any one of the individual profiles that are combined, and shall continue to require their implementation. Similarly, conditions for deprecated conditional classes shall apply (as stated in the support for conditional classes).

- **Support for required properties:** A combination of SMI-S profiles shall support all required properties of classes in any one of the individual profiles that are combined and shall continue to require them. If a property is required in any of the individual profiles, then the property will be required in the combined profile.
- **Support for conditional properties:** A combination of SMI-S profiles shall support all conditional properties of classes in the individual profiles that are combined and shall continue to require them as specified by the conditions of the individual profiles that are combined. If a property is conditional in one or more of the individual profiles (and not required in any other individual profile) then it shall be conditional in the combination profile. If a property is conditional in multiple individual profiles, but with different conditions, then all conditions shall yield the existence of the class.
- **Deprecation of properties:** A combination of SMI-S profiles may include deprecated (via the CIM schema) properties of classes introduced in any one of the individual profiles that are combined, and shall continue to require their implementation. Similarly, conditions for deprecated conditional properties shall apply (as stated in the support for conditional properties).
- **Support for component profiles:** A combination of SMI-S profiles shall support the functionality of all component profiles of all of the individual profiles that are combined and shall continue to require them if they were required in any one of the individual profiles that are combined. If a combination of SMI-S profiles results in two references to a component profile by the same name from multiple individual profiles that were combined, the combined profile may require multiple implementations if the component profiles in question have different major version numbers. And if the component profiles have different minor version numbers, then the higher version number shall be implemented (since it provides backward compatibility to the earlier component profile).

If a component profile is required in any one of the individual profiles then it will be required in the combined profile.

If a component profile is not required in any of the individual profiles, but is conditional in at least one of the individual profiles, then it will be conditional in the combined profile. If a component profile is conditional in multiple individual profiles (that are being combined) then all conditions shall yield existence of the component profile.

B.6.3 Conditions for a New Profile

If any of the conditions outlined in B.6 Rules for Combining (Autonomous) Profiles cannot be satisfied, then a new profile shall be defined that represents the desired semantic of the device (or program) in question.

EXPERIMENTAL

B.7 Rules for Vendor Extensions

SMI-S is intended to be extended by vendor implementations to cover vendor function that is not covered by SMI-S. Such extensions allow clients to exploit vendor functions that are not covered by SMI-S, when the client has awareness of the specific functions of the implementation. However, the extensions need to be done in such a way that they do not cause clients that support the functions in SMI-S to fail. This clause describes the rules for doing vendor unique extensions to SMI-S.

B.7.1 Objectives for Vendor Extension Rules

The basic objectives for the rules associated with vendor extensions are:

- Vendor extensions shall follow the compliance rules (as defined in B.3).
- Vendor extensions shall follow the backward compatibility rules (as defined in B.5).

- Vendor extensions shall avoid extensions that nullify the existing SMI-S.
- Vendor extensions shall avoid extensions that would confuse clients.

B.7.2 Vendor Extensions and Compliance Rules

When implementing a vendor extension, the following rules shall be followed:

- When an implementation claims compliance to an SMI-S profile (RegisteredOrganization="SNIA") the implementation shall honor the behavior for CIM Elements and methods as outlined in the Profile.
 - All CIM Elements (Classes, properties and methods) defined by the SNIA profile shall be honored, including mandatory and conditional elements.

For example, if an SMI-S Profile defines a StorageVolume class with mandatory properties, a vendor extension to the profile may not define a StorageVolume that has fewer mandatory properties.
 - Similarly, if the StorageVolume class is mandatory, a vendor extension may not render the use of the class as conditional (or optional).
- Instances of CIM associations between CIM Instances shall exist as defined by the profile.
 - A vendor extension may add associations, but the mandatory and conditional associations between instances of a class specified by the profile shall exist.

For example, if an SMI-S profile defines a mandatory DeviceSAPImplementation association between a ProtocolEndpoint and a LogicalPort, a vendor extension that adds a new ProtocolEndpoint shall also have the DeviceSAPImplementation association.
 - If this is not reasonable for the extension, for whatever reason, the vendor extension should consider using different classes.
- When a vendor extension uses a superclass of a given CIM class used in the SMI-S profile, the extension shall honor the attributes and behaviors defined for the superclass.
 - If a vendor extension uses "System" in an SMI-S Profile that defines "ComputerSystem" classes, the extension shall honor properties and associations of the inherited from System defined in the SMI-S profile.

B.7.3 Vendor Extensions and Backward Compatibility Rules

When the backward compatibility rules are applied to vendor extended SMI-S profiles, the following rules shall apply:

- Vendor extensions that are deployed in a customer environment shall work with existing SMI-S Clients and SMI-S Clients that are not aware of the extensions.

This includes:

- SMI-S operations continue to work against the extended profile implementation.
- SMI-S Clients can support a given profile version or extended versions of a given profile.
- A vendor extended implementation of an SMI-S Profile shall be backward compatible to the SMI-S profile.
- If an extended version of an SMI-S profile cannot maintain backward compatibility to the SMI-S profile, it shall be defined as a different profile (e.g., RegisteredOrganization="VendorID"). Otherwise the client may assume that the extended profile is backwards compatible and that all operations on the SMI-S profile will continue to work in the extended version of the profile.

B.7.4 Vendor Extensions and SMI-S Nullification

Vendor extensions shall avoid extensions that nullify an existing SMI-S profile. This includes, but is not limited to:

- Adding Classes that the implementation considers mandatory such that a client will fail if it does not establish instances of the class.

This does not mean that such classes cannot be provided by the provider implementation. But it cannot expect an SMI-S client to overtly create such class instances.

- Adding class properties that an implementation considers mandatory such that a client will fail if it does not set values for such properties

For example, if a `SettingData` is used as a parameter of an SMI-S extrinsic method, a vendor extension cannot extend the `SettingData` with a mandatory property such that it will fail the client if it does not set the extended property.

An implementation may extend a `SettingData` class that is used in an SMI-S extrinsic method if the implementation supports a default value for the property.

- A vendor extension shall not extend the method signature of any SMI-S extrinsic method.

If a vendor wants to define an extended version of an SMI-S method, it should define a new method with the extended parameter list.

Ideally, the vendor extension should support the SMI-S method, however, this is not required if the profile has a `capabilities` class that identifies whether or not the method in question is supported.

- A vendor extension should not extend the definition of a property unless the SMI-S profile makes provisions for “vendor extensions” to the property.

For example, properties that are enumerations with a “vendor extension” range that is formally recognized in the SMI-S specification may be extended (vendor extension enumerations added).

However, if a property that is an enumeration, but SMI-S does not formally recognize that a “vendor extension” range of the enumeration, then the vendor extension should not use the property for this.

B.7.5 Vendor Extensions that Avoid Client Confusion

Vendor extensions shall avoid extensions that would confuse clients. This includes, but is not limited to:

- Reuse of Classes used by SMI-S profiles should be avoided (e.g., vendor extended usages).

For example, if an SMI-S autonomous profile defines four uses of `StorageExtent`, it would be unfortunate if a vendor extension defined a fifth usage of `StorageExtent`. The general SMI-S client will be looking for four different uses of `StorageExtent` and will likely get confused by the fifth usage.

To avoid this, the vendor extension should define a new class (with all the `StorageExtent` properties that apply).

- Use of CIM properties not specified by SMI-S

There are several properties “not specified” by SMI-S, but are included in CIM MOFs. An example might be `Caption` on `ManagedElement`. Vendor extensions should avoid using CIM properties for a specific purpose defined by the vendor extension.

Future versions of SMI-S may, in fact, define a specific use of the CIM property and the chances that it matches the use in every vendor extension is somewhat unlikely.

It is safer if the vendor extension defines its own unique properties rather than attempt to re-use CIM properties.

EXPERIMENTAL
