



1

Document Number: DSP1021

2

Date: 2009-06-16

3

4

Version: 1.0.0

5 **Shared Device Management Profile**

6 **Document Type: Specification**

7 **Document Status: DMTF Standard**

8 **Document Language: E**

9 Copyright Notice

10 Copyright © 2006, 2009 Distributed Management Task Force, Inc. (DMTF). All rights reserved.

11 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
12 management and interoperability. Members and non-members may reproduce DMTF specifications and
13 documents, provided that correct attribution is given. As DMTF specifications may be revised from time to
14 time, the particular version and release date should always be noted.

15 Implementation of certain elements of this standard or proposed standard may be subject to third party
16 patent rights, including provisional patent rights (herein "patent rights"). DMTF makes no representations
17 to users of the standard as to the existence of such rights, and is not responsible to recognize, disclose,
18 or identify any or all such third party patent right, owners or claimants, nor for any incomplete or
19 inaccurate identification or disclosure of such rights, owners or claimants. DMTF shall have no liability to
20 any party, in any manner or circumstance, under any legal theory whatsoever, for failure to recognize,
21 disclose, or identify any such third party patent rights, or for such party's reliance on the standard or
22 incorporation thereof in its product, protocols or testing procedures. DMTF shall have no liability to any
23 party implementing such standard, whether such implementation is foreseeable or not, nor to any patent
24 owner or claimant, and shall have no liability or responsibility for costs or losses incurred if a standard is
25 withdrawn or modified after publication, and shall be indemnified and held harmless by any party
26 implementing the standard from any and all claims of infringement by a patent owner for such
27 implementations.

28 For information about patents held by third-parties which have notified the DMTF that, in their opinion,
29 such patent may relate to or impact implementations of DMTF standards, visit
30 <http://www.dmtf.org/about/policies/disclosures.php>.

31

CONTENTS

32 Foreword 5

33 Introduction 6

34 1 Scope 7

35 2 Normative References..... 7

36 2.1 Approved References 7

37 2.2 Other References..... 7

38 3 Terms and Definitions 7

39 4 Symbols and Abbreviated Terms 9

40 5 Synopsis 9

41 6 Description 9

42 7 Implementation Requirements 11

43 7.1 Rules for Instrumenting Shared Devices 11

44 7.2 State Management of SharedDeviceManagementService..... 12

45 8 Methods..... 14

46 8.1 CIM_SharedDeviceManagementService.ShareDevice() 14

47 8.2 CIM_SharedDeviceManagementService.RequestStateChange() (Conditional) 16

48 8.3 Profile Conventions for Operations 17

49 8.4 CIM_DeviceSharingCapabilities 17

50 8.5 CIM_ElementCapabilities 17

51 8.6 CIM_EnabledLogicalElementCapabilities..... 18

52 8.7 CIM_HostedService 18

53 8.8 CIM_LogicalDevice 18

54 8.9 CIM_ServiceAffectsElement 19

55 8.10 CIM_SharedDeviceManagementService..... 19

56 8.11 CIM_SharingDependency..... 20

57 8.12 CIM_SystemDevice 20

58 9 Use Cases 20

59 9.1 Object Diagrams 21

60 9.2 Determining If a Logical Device Is a Shared Device 22

61 9.3 Finding the CIM_LogicalDevice Instance for a Real Device..... 22

62 9.4 Determining How a Device Can Be Shared 23

63 9.5 Determining System Access to a Shared Device 23

64 9.6 Changing Shared Device Access 24

65 9.7 Determining If ElementName Can Be Modified..... 24

66 10 CIM Elements 24

67 10.1 CIM_DeviceSharingCapabilities 25

68 10.2 CIM_ElementCapabilities – SharingCapabilities 25

69 10.3 CIM_ElementCapabilities – EnabledLogicalElementCapabilities 25

70 10.4 CIM_EnabledLogicalElementCapabilities..... 25

71 10.5 CIM_HostedService 26

72 10.6 CIM_LogicalDevice 26

73 10.7 CIM_RegisteredProfile..... 26

74 10.8 CIM_ServiceAffectsElement 27

75 10.9 CIM_SharedDeviceManagementService..... 27

76 10.10 CIM_SharingDependency..... 27

77 10.11 CIM_SystemDevice 28

78 ANNEX A (informative) Change Log..... 29

79

80 Figures

81	Figure 1 – Shared Device Management Profile: Class Diagram	10
82	Figure 2 – Shared Device Management	21
83	Figure 3 – Registered Profile	22

84

85 Tables

86	Table 1 – Referenced Profiles	9
87	Table 2 – CIM_SharedDeviceManagementService.ShareDevice() Method: Return Code Values.....	14
88	Table 3 – CIM_SharedDeviceManagementService.ShareDevice() Method: Parameters.....	15
89	Table 4 – CIM_SharedDeviceManagementService.RequestStateChange() Method: Return Code	
90	Values	16
91	Table 5 – CIM_SharedDeviceManagementService.RequestStateChange() Method: Parameters.....	16
92	Table 6 – CIM_DeviceSharingCapabilities Operations.....	17
93	Table 7 – CIM_ElementCapabilities Operations	18
94	Table 8 – CIM_EnabledLogicalElementCapabilities Operations	18
95	Table 9 – CIM_HostedService Operations	18
96	Table 10 – CIM_LogicalDevice Operations	19
97	Table 11 – CIM_ServiceAffectsElement Operations	19
98	Table 12 – CIM_HostedService Operations	19
99	Table 13 – CIM_SharingDependency Operations	20
100	Table 14 – CIM_SystemDevice Operations.....	20
101	Table 15 – Required CIM Elements: Shared Device Management Profile.....	24
102	Table 16 – Class: CIM_DeviceSharingCapabilities	25
103	Table 17 – Class: CIM_ElementCapabilities – SharingCapabilities	25
104	Table 18 – Class: CIM_ElementCapabilities – EnabledLogicalElementCapabilities.....	25
105	Table 19 – Class: CIM_EnabledLogicalElementCapabilities.....	25
106	Table 20 – Class: CIM_HostedService	26
107	Table 21 – Class: CIM_LogicalDevice	26
108	Table 22 – Class: CIM_RegisteredProfile.....	26
109	Table 23 – Class: CIM_ServiceAffectsElement	27
110	Table 24 – Class: CIM_SharedDeviceManagementService.....	27
111	Table 25 – Class: CIM_SharingDependency.....	27
112	Table 26 – Class: CIM_SystemDevice	28

113

114

Foreword

115 The *Shared Device Management Profile* (DSP1021) was prepared by the Server Management Working
116 Group and the Physical Platform Profiles Working Group.

117 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
118 management and interoperability.

119 Acknowledgments

120 The authors wish to acknowledge the following people.

121 Editor:

- 122 • Aaron Merkin – IBM

123 Contributors:

- 124 • Jon Hass – Dell
- 125 • Khachatur Papanyan – Dell
- 126 • Enoch Suen – Dell
- 127 • Jeff Hilland – HP
- 128 • Christina Shaw – HP
- 129 • Aaron Merkin – IBM
- 130 • Perry Vincent – Intel
- 131 • John Leung – Intel

132

133

Introduction

134 The information in this specification should be sufficient for a provider or consumer of this data to identify
135 unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to
136 represent and manage shared devices of a modular system that is modeled using the DMTF CIM core
137 and extended model definitions. The target audience for this specification is implementers who are writing
138 CIM-based providers or consumers of management interfaces that represent the component described in
139 this document.

140

Shared Device Management Profile

141 1 Scope

142 The *Shared Device Management Profile* is a component profile for modeling shared devices of modular
143 systems.

144 2 Normative References

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

148 2.1 Approved References

149 DMTF DSP0004, *CIM Infrastructure Specification 2.3*,
150 http://www.dmtf.org/standards/published_documents/DSP0004_2.3.pdf

151 DMTF DSP0200, *CIM Operations over HTTP 1.2*,
152 http://www.dmtf.org/standards/published_documents/DSP0200_1.2.pdf

153 DMTF DSP1001, *Management Profile Specification Usage Guide 1.0*,
154 http://www.dmtf.org/standards/published_documents/DSP1001_1.0.pdf

155 DMTF DSP1018, *Service Processor Profile 1.0*,
156 http://www.dmtf.org/standards/published_documents/DSP1018_1.0.pdf

157 DMTF DSP1033, *Profile Registration Profile 1.0*,
158 http://www.dmtf.org/standards/published_documents/DSP1033_1.0.pdf

159 2.2 Other References

160 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
161 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>

162 3 Terms and Definitions

163 For the purposes of this document, the following terms and definitions apply.

164 3.1

165 **can**

166 used for statements of possibility and capability, whether material, physical, or causal

167 3.2

168 **cannot**

169 used for statements of possibility and capability, whether material, physical, or causal

170 3.3

171 **conditional**

172 indicates requirements to be followed strictly to conform to the document when the specified conditions
173 are met

- 174 **3.4**
175 **mandatory**
176 indicates requirements to be followed strictly to conform to the document and from which no deviation is
177 permitted
- 178 **3.5**
179 **may**
180 indicates a course of action permissible within the limits of the document
- 181 **3.6**
182 **need not**
183 indicates a course of action permissible within the limits of the document
- 184 **3.7**
185 **optional**
186 indicates a course of action permissible within the limits of the document
- 187 **3.8**
188 **referencing profile**
189 indicates a profile that owns the definition of this class and can include a reference to this profile in its
190 "Related Profiles" table
- 191 **3.9**
192 **shall**
193 indicates requirements to be followed strictly to conform to the document and from which no deviation is
194 permitted
- 195 **3.10**
196 **shall not**
197 indicates requirements to be followed strictly to conform to the document and from which no deviation is
198 permitted
- 199 **3.11**
200 **should**
201 indicates that, among several possibilities, one is recommended as particularly suitable, without
202 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 203 **3.12**
204 **should not**
205 indicates that a certain possibility or course of action is deprecated but not prohibited
- 206 **3.13**
207 **Real Logical Device**
208 the instance of CIM_LogicalDevice that represents the underlying shared device
- 209 **3.14**
210 **Sharing Logical Device**
211 an instance of CIM_LogicalDevice that represents a Sharing System's view of the underlying shared
212 resource
- 213 **3.15**
214 **Sharing System**
215 an instance of CIM_ComputerSystem that represents a system that uses a shared device

216 **3.16**
 217 **Owning System**
 218 an instance of CIM_ComputerSystem that represents the system that owns the shared resource

219 **4 Symbols and Abbreviated Terms**

220 **4.1**
 221 **CIM**
 222 Common Information Model

223 **4.2**
 224 **MOF**
 225 Managed Object Format

226 **5 Synopsis**

227 **Profile Name:** Shared Device Management

228 **Version:** 1.0.0

229 **Organization:** DMTF

230 **CIM Schema version:** 2.22

231 **Central Class:** CIM_SharedDeviceManagementService

232 **Scoping Class:** CIM_ComputerSystem

233 The *Shared Device Management Profile* extends management capability to include support for managing
 234 shared devices of a modular system. This includes support for modeling shared devices and the logical
 235 containment hierarchy.

236 The Scoping Instance of the *Shared Device Management Profile* shall be the instance of
 237 CIM_ComputerSystem with which the Central Instance is associated through an instance of the
 238 CIM_HostedService association.

239 Table 1 identifies profiles on which this profile has a dependency.

240 **Table 1 – Referenced Profiles**

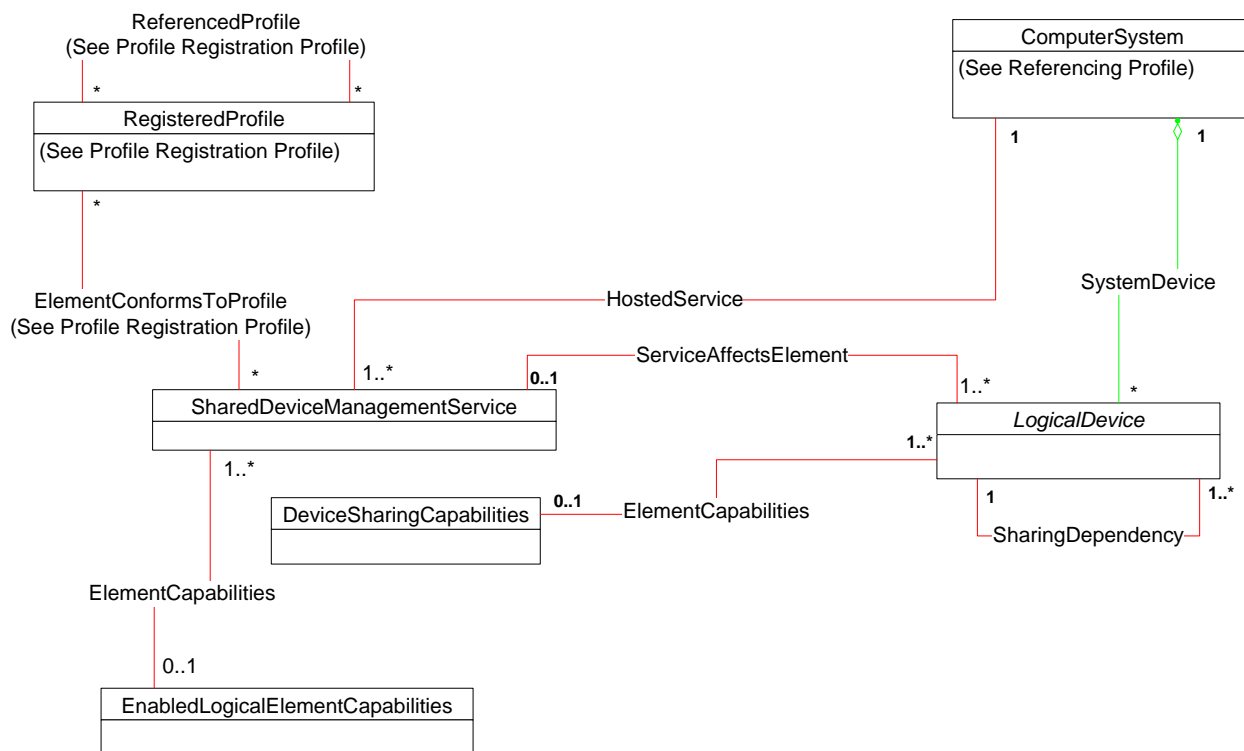
Profile Name	Organization	Version	Relationship	Behavior
Profile Registration	DMTF	1.0	Mandatory	None

241 **6 Description**

242 The *Shared Device Management Profile* describes management of shared devices of a modular system.
 243 Some blade server systems that host server blades provide media devices within the enclosure that can
 244 be accessed by the server blades. Some systems will allow concurrent access and others will support
 245 access only from a single server blade at a time. A management controller is generally responsible for
 246 arbitrating and managing access to the shared devices. The *Shared Device Management Profile* contains
 247 the CIM elements necessary to manage shared devices of a modular system. Its scope is limited to
 248 defining those classes or behaviors that are unique to the management of shared devices.

249 Figure 1 represents the class schema for the *Shared Device Management Profile*. For simplicity, the
 250 prefix CIM_ has been removed from the names of the classes.

251



252

253

Figure 1 – Shared Device Management Profile: Class Diagram

254 As indicated in Figure 1, the *Shared Device Management Profile* includes the following classes:
 255 CIM_SharedDeviceManagementService, CIM_DeviceSharingCapabilities, CIM_ServiceAffectsElement,
 256 CIM_SharingDependency, CIM_LogicalDevice, CIM_ElementCapabilities, and
 257 CIM_EnabledLogicalElementCapabilities. The CIM_SharedDeviceManagementService represents the
 258 ability to manage physical access to a shared device. CIM_DeviceSharingCapabilities indicates the ability
 259 of the client system to access the shared device. This profile creates an additional instance of
 260 CIM_LogicalDevice to represent the client system's view of the shared device. CIM_SharingDependency
 261 represents the relationship between this CIM_LogicalDevice instance and the CIM_LogicalDevice
 262 instance that represents the underlying shared resource. For examples of how to use these classes, see
 263 section 8.11.1. For a complete list of classes, see section 10.

264 The specification describes two ways that instances of CIM_LogicalDevice are used.

265 The Real Logical Device identifies an instance of a subclass of CIM_LogicalDevice that represents the
 266 actual device being shared. It is expected that this instance will be instrumented in accordance with the
 267 profile or profiles appropriate for the device type.

268 A Sharing Logical Device is an instance of the same subclass of CIM_LogicalDevice as that of the Real
 269 Logical Device. The *Shared Device Management Profile* uses this instance to provide a manageability
 270 point for system access to a shared device. Therefore, the specification of the Sharing Logical Device will
 271 be owned by this profile specification. A Sharing Logical Device instance does not correspond to a distinct
 272 real-world device. Therefore, it is not intended to be used as a focal point for any device management
 273 beyond managing access to that device for a Sharing System.

274 In Figure 1, CIM_LogicalDevice represents both a Real Logical Device and a Sharing Logical Device. The
 275 CIM_ElementCapabilities association references CIM_LogicalDevice because it represents a Sharing

276 Logical Device. The CIM_ServiceAffectsElement association references the CIM_LogicalDevice instance
277 because it represents a Real Logical Device.

278 **7 Implementation Requirements**

279 This section describes the implementation requirements of the *Shared Device Management Profile*.
280 Required methods are listed in section 8, and properties are listed in section 10.

281 **7.1 Rules for Instrumenting Shared Devices**

282 This section describes the requirements for instrumenting Shared Devices.

283 **7.1.1 Instrumenting the Sharing Logical Device**

284 Each representation of the shared device in a sharing system shall have one instance of
285 CIM_LogicalDevice. These instances are identified as the Sharing Logical Device instances. Each
286 Sharing Logical Device shall have exactly one instance of CIM_SystemDevice associating a Sharing
287 Logical Device instance to a Sharing System. The Sharing Logical Device shall be an instance of the
288 same subclass of CIM_LogicalDevice as that of the Real Logical Device.

289 **7.1.2 Capabilities**

290 CIM_DeviceSharingCapabilities indicates the capabilities of a client system to use a shared device. An
291 instance of CIM_SharingCapabilities associated with the Sharing Logical Device instance shall indicate
292 the capabilities of the associated Sharing System to use the underlying shared device. The underlying
293 shared device is represented by the Real Logical Device instance associated with the Sharing Logical
294 Device through an instance of CIM_SharingDependency.

295 Exactly one instance of the CIM_ElementCapabilities shall associate a Sharing Logical Device instance
296 with an instance of CIM_SharingCapabilities.

297 **7.1.2.1 CIM_ElementCapabilities.ManagedElement**

298 The Sharing Logical Device instance shall be the value of the CIM_ElementCapabilities.ManagedElement
299 reference.

300 **7.1.2.2 CIM_ElementCapabilities.Capabilities**

301 An instance of CIM_SharingCapabilities shall be the value of the Capabilities reference.

302 **7.1.3 CIM_ServiceAffectsElement**

303 At least one instance of CIM_ServiceAffectsElement shall associate an instance of
304 CIM_SharedDeviceManagementService with each Real Logical Device.

305 **7.1.3.1 CIM_ServiceAffectsElement.ElementAffects**

306 The ElementAffects property of the CIM_ServiceAffectsElement instance shall have a value of 5
307 (Manages).

308 **7.1.3.2 CIM_ServiceAffectsElement.UserOfService**

309 The UserOfService property shall reference the Real Logical Device.

310 **7.1.4 CIM_SharingDependency**

311 Exactly one instance of CIM_SharingDependency shall associate a Sharing Logical Device instance with
312 the Real Logical Device instance.

313 **7.1.4.1 CIM_SharingDependency.Dependent**

314 The reference to the Sharing Logical Device shall be the value of the Dependent property.

315 **7.1.4.2 CIM_SharingDependency.Antecedent**

316 The reference to the Real Logical Device shall be the value of the Antecedent property.

317 A reference to a Sharing Logical Device shall not be the value of the Antecedent property of any instance
318 of CIM_SharingDependency.

319 **7.1.5 CIM_Realizes**

320 A reference to a Sharing Logical Device shall not be the value of the Dependent property of any instance
321 of CIM_Realizes.

322 **7.2 State Management of SharedDeviceManagementService**

323 This section describes the requirements when the optional behavior of managing the state of the Shared
324 Device Management Service is implemented or not implemented.

325 **7.2.1 Shared Device Management Service State Management Is Supported (Conditional)**

326 When management of the state of a Shared Device Management Service is supported, exactly one
327 instance of CIM_EnabledLogicalElementCapabilities shall be associated with the
328 CIM_SharedDeviceManagementService instance through an instance of CIM_ElementCapabilities.

329 Support for managing the state of the port Shared Device Management Service is optional behavior. This
330 section describes the CIM elements and behaviors that shall be implemented when this optional behavior
331 is supported.

332 **Conditional Determination:** A client can determine whether state management is supported as follows:

- 333 1) Find the CIM_EnabledLogicalElementCapabilities instance associated with the
334 CIM_SharedDeviceManagementService instance.
- 335 2) Query the value of the RequestedStatesSupported property. If at least one value is specified,
336 state management is supported.

337 **7.2.1.1 CIM_EnabledLogicalElementCapabilities**

338 When state management is supported, exactly one instance of CIM_EnabledLogicalElement capabilities
339 shall be associated with the CIM_SharedDeviceManagementService instance through an instance of the
340 CIM_ElementCapabilities association.

341 **7.2.1.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

342 The RequestedStatesSupported property may contain zero or more of the following values: 2 (Enabled),
343 3 (Disabled), or 11 (Reset).

344 **7.2.1.2 CIM_SharedDeviceManagementService.RequestedState**

345 When the CIM_SharedDeviceManagementService.RequestStateChange() method is successfully
346 invoked, the value of the RequestedState property shall be the value of the RequestedState parameter. If
347 the method is not successfully invoked, the value of the RequestedState property is indeterminate.

348 The CIM_SharedDeviceManagementService.RequestedState property shall have one of the values
349 specified in the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property or 5 (No
350 Change).

351 **7.2.1.3 CIM_SharedDeviceManagementService.EnabledState**

352 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled) and the
353 CIM_SharedDeviceManagementService.RequestStateChange() method completes successfully, the
354 value of the EnabledState property shall equal the value of the
355 CIM_SharedDeviceManagementService.RequestedState property.

356 If the method does not complete successfully, the value of the EnabledState property is indeterminate.

357 The EnabledState property shall have the value 2 (Enabled), 3 (Disabled), or 6 (Enabled but Offline).

358 **7.2.2 Shared Device Management Service State Management Is Not Supported**

359 This section describes the CIM elements and behaviors that shall be implemented when management of
360 the Shared Device Management Service state is not supported.

361 **7.2.2.1 CIM_EnabledLogicalElementCapabilities**

362 When state management is not supported, exactly one instance of CIM_EnabledLogicalElement
363 capabilities may be associated with the CIM_SharedDeviceManagementService instance through an
364 instance of the CIM_ElementCapabilities association.

365 **7.2.2.1.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

366 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall not contain any
367 values.

368 **7.2.2.2 CIM_SharedDeviceManagementService.RequestedState**

369 The RequestedState property shall have the value 12 (Not Applicable).

370 **7.2.2.3 CIM_SharedDeviceManagementService.EnabledState**

371 The EnabledState property shall have one of the following values: 2 (Enabled), 3 (Disabled), 5 (Not
372 Applicable), or 6 (Enabled but Offline).

373 **7.2.3 Modifying ElementName Is Supported (Conditional)**

374 The CIM_SharedDeviceManagementService.ElementName property may support being modified by the
375 ModifyInstance operation. See section 8.10.1.1. This is conditional behavior. This section describes the
376 CIM elements and behavior requirements when an implementation supports client modification of the
377 CIM_SharedDeviceManagementService.ElementName property.

378 **7.2.3.1 CIM_EnabledLogicalElementCapabilities**

379 An instance of CIM_EnabledLogicalElementCapabilities shall be associated with the
380 CIM_SharedDeviceManagementService instance through an instance of CIM_ElementCapabilities.

381 **7.2.3.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

382 This property shall have a value of TRUE when the implementation supports client modification of the
383 CIM_SharedDeviceManagementService.ElementName property.

384 7.2.3.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

385 The MaxElementNameLen property shall be implemented.

386 7.2.4 Modifying ElementName Is Not Supported

387 This section describes the CIM elements and behaviors that shall be implemented when the
388 CIM_SharedDeviceManagementService.ElementName does not support being modified by the
389 ModifyInstance operation.

390 7.2.4.1 CIM_EnabledLogicalElementCapabilities

391 An instance of CIM_EnabledLogicalElementCapabilities may be associated with the
392 CIM_SharedDeviceManagementService instance through an instance of CIM_ElementCapabilities.

393 7.2.4.1.1 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

394 This property shall have a value of FALSE when the implementation does not support client modification
395 of the CIM_SharedDeviceManagementService.ElementName property.

396 7.2.4.1.2 CIM_EnabledLogicalElement.MaxElementNameLen

397 The MaxElementNameLen property may be implemented. The MaxElementNameLen property is
398 irrelevant in this context.

399 8 Methods

400 This section details the requirements for supporting intrinsic operations and extrinsic methods for the CIM
401 elements defined by this profile.

402 8.1 CIM_SharedDeviceManagementService.ShareDevice()

403 The ShareDevice() method defines the ability to change the access of a system to a shared device.
404 When this method completes successfully, the value of the CurrentAccess property of the instance of
405 CIM_SharingDependency that associates the Real Logical Device identified by the Device parameter with
406 the Sharing Logical Device that is scoped to the CIM_ComputerSystem instance identified by the System
407 parameter shall have the value of the RequestedAccess parameter to the method.

408 Detailed requirements of the ShareDevice() method are specified in Table 2 and Table 3.

409 No standard messages are defined.

410 **Table 2 – CIM_SharedDeviceManagementService.ShareDevice() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
1	Method is not supported in implementation.
2	Unknown or unspecified error

411 **Table 3 – CIM_SharedDeviceManagementService.ShareDevice() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN	RequestedAccess	uint16	The desired access
IN	Device	CIM_LogicalDevice REF	The Real Logical Device to change access to
IN	System	CIM_System REF	System for which access to the shared device is being modified
IN	TimeoutPeriod	datetime	The maximum amount of time to wait for the modification to occur
IN	Force	Boolean	Whether to forcibly revoke access from the current owner if necessary

412 If the requested access to the shared device is granted, the method shall return 0 (zero).

413 If the TimeoutPeriod parameter specifies a valid interval and the requested access cannot be granted in
414 the interval specified, the method shall return 2.

415 If the operation fails and a more specific error is not applicable, the method shall return 2.

416 The method shall return 2 if one or more of the following conditions are met:

- 417 • The TimeoutPeriod parameter is supported by the implementation, the value is specified (that
418 is, not null), and the interval is not valid.
- 419 • The Device parameter is null.
- 420 • The System parameter is null.
- 421 • No instance of CIM_SystemDevice associates the CIM_LogicalDevice instance identified by the
422 Device parameter with the CIM_System instance identified by the System parameter.

423 The method shall return 2 if the RequestedAccess parameter value is not one of the values in the
424 SupportedAccessModes property of the instance of CIM_DeviceSharingCapabilities that is associated
425 through an instance of ElementCapabilities to the CIM_LogicalDevice instance identified by the Device
426 parameter to the method.

427 The method shall return 2 if the CIM_LogicalDevice instance identified by the Device parameter is not
428 associated with a CIM_LogicalDevice instance that is associated with this instance of
429 CIM_SharedDeviceManagementService through an instance of CIM_ServiceAffectsElement.

430 The method shall return 2 if the CIM_System instance identified by the System parameter is not
431 associated with an instance of a sharing CIM_LogicalDevice. The CIM_LogicalDevice instance is
432 associated with an instance of a real CIM_LogicalDevice that is associated with this instance of
433 CIM_SharedDeviceManagementService through an instance of CIM_ServiceAffectsElement.

434 The method shall return 2 if the value of the Timeout parameter is not null and the implementation does
435 not support this parameter following the specification in the MOF.

436 The method shall return 2 if the target device is currently in use and cannot be reassigned. The method
437 shall validate whether the requested access can be granted without interfering with access already
438 granted to the device for another sharing system. If the value of the Force parameter is FALSE, or the
439 Force parameter is not specified and the requested access cannot be granted without interfering with a
440 different system's existing access to the shared device, the method shall return 2 and not modify access
441 to the shared device.

442 8.2 CIM_SharedDeviceManagementService.RequestStateChange() (Conditional)

443 Invocation of the CIM_SharedDeviceManagementService.RequestStateChange() method changes the
 444 element's state to the value specified in the RequestedState parameter. The Enabled or Disabled values
 445 of the RequestedState parameter shall correspond to enabling or disabling the functionality represented
 446 by the instance of CIM_SharedDeviceManagementService. A value of 2 (Enabled) shall correspond to a
 447 request to enable the functionality. A value of 3 (Disabled) shall correspond to a request to disable the
 448 functionality.

449 See section 7.2.1.2 for information about the effect of this method on the RequestedState property.

450 When the RequestedState parameter has a value of 2 (Enabled) or 3 (Disabled), the method shall be
 451 considered successful when completed if the availability of the functionality upon completion corresponds
 452 to the desired availability indicated by the RequestedState parameter. An actual change in state is not
 453 necessary for the method to be considered successful as long as the resultant state is equal to the
 454 requested state. Upon successful completion of the method, the return value shall be 0 (zero).

455 See section 7.2.1.3 for information about the effect of this method on the EnabledState property.

456 Detailed requirements of the RequestStateChange() method are specified in Table 4 and Table 5.

457 No standard messages are defined.

458 Invoking the CIM_SharedDeviceManagementService.RequestStateChange() method multiple times
 459 could result in earlier requests being overwritten or lost.

460 **Table 4 – CIM_SharedDeviceManagementService.RequestStateChange() Method: Return Code**
 461 **Values**

Value	Description
0	Request was successfully executed.
1	Method is not supported in the implementation.
2	Error occurred.
0x1000	Job started: REF returned to started CIM_ConcreteJob

462 **Table 5 – CIM_SharedDeviceManagementService.RequestStateChange() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Valid state values : 2 (Enabled) 3 (Disabled) 11 (Reset)
OUT	Job	CIM_ConcreteJob REF	Returned if job started
IN, REQ	TimeoutPeriod	datetime	Client-specified maximum amount of time allowed for the transition to a new state: 0 or NULL – No time requirements <interval> – Maximum time allowed

463 **8.2.1 CIM_SharedDeviceManagementService.RequestStateChange() Conditional**
 464 **Support**

465 The CIM_SharedDeviceManagementService.RequestStateChange() method shall be supported and
 466 shall not return a value of 1 (Not Supported) when an instance of
 467 CIM_EnabledLogicalElementCapabilities is associated with the CIM_SharedDeviceManagementService
 468 instance and the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property contains
 469 at least one value.

470 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
 471 CIM_SharedDeviceManagementService, the
 472 CIM_SharedDeviceManagementService.RequestStateChange() method shall not be implemented.

473 **8.3 Profile Conventions for Operations**

474 For each profile class (including associations), the implementation requirements for operations, including
 475 those in the following default list, are specified in class-specific subclauses of this clause.

476 The default list of operations is as follows:

- 477 • GetInstance
- 478 • Associators
- 479 • AssociatorNames
- 480 • References
- 481 • ReferenceNames
- 482 • EnumerateInstances
- 483 • EnumerateInstanceNames

484 **8.4 CIM_DeviceSharingCapabilities**

485 Table 6 lists implementation requirements for operations. If implemented, these operations shall be
 486 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 6, all operations in
 487 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

488 NOTE: Related profiles may define additional requirements on operations for the profile class.

489 **Table 6 – CIM_DeviceSharingCapabilities Operations**

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.4.1.	None

490 **8.4.1 ModifyInstance**

491 When the ModifyInstance operation is supported for an instance of CIM_DeviceSharingCapabilities, the
 492 ModifyInstance operation shall not modify the SupportedAccessModes property.

493 **8.5 CIM_ElementCapabilities**

494 Table 7 lists implementation requirements for operations. If implemented, these operations shall be
 495 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 7, all operations in
 496 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

497 NOTE: Related profiles may define additional requirements on operations for the profile class.

498

Table 7 – CIM_ElementCapabilities Operations

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

499 8.6 CIM_EnabledLogicalElementCapabilities

500 Table 8 lists implementation requirements for operations. If implemented, these operations shall be
 501 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 8, all operations in
 502 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

503 NOTE: Related profiles may define additional requirements on operations for the profile class.

504

Table 8 – CIM_EnabledLogicalElementCapabilities Operations

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.6.1.	None

505 8.6.1 ModifyInstance

506 When the ModifyInstance operation is supported for an instance of
 507 CIM_EnabledLogicalElementCapabilities, the ModifyInstance operation shall not modify the following
 508 properties:

- 509 • RequestedStatesSupported
- 510 • ElementNameEditSupported
- 511 • MaxElementNameLen

512 8.7 CIM_HostedService

513 Table 9 lists implementation requirements for operations. If implemented, these operations shall be
 514 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 9, all operations in
 515 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

516 NOTE: Related profiles may define additional requirements on operations for the profile class.

517

Table 9 – CIM_HostedService Operations

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

518 8.8 CIM_LogicalDevice

519 Table 10 lists implementation requirements for operations. If implemented, these operations shall be
 520 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations
 521 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

522 NOTE: Related profiles may define additional requirements on operations for the profile class.

523 **Table 10 – CIM_LogicalDevice Operations**

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.8.1.	None

524 **8.8.1 ModifyInstance**

525 When the ModifyInstance operation is supported for an instance of CIM_LogicalDevice, the
 526 ModifyInstance operation shall support modification of the ElementName property.

527 **8.9 CIM_ServiceAffectsElement**

528 Table 11 lists implementation requirements for operations. If implemented, these operations shall be
 529 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 11, all operations
 530 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

531 NOTE: Related profiles may define additional requirements on operations for the profile class.

532 **Table 11 – CIM_ServiceAffectsElement Operations**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

533 **8.10 CIM_SharedDeviceManagementService**

534 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
 535 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 12, all operations
 536 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

537 NOTE: Related profiles may define additional requirements on operations for the profile class.

538 **Table 12 – CIM_HostedService Operations**

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.10.1.	None

539 **8.10.1 CIM_SharedDeviceManagementService – ModifyInstance Operation**

540 This section details the specific requirements for the ModifyInstance operation applied to an instance of
 541 CIM_SharedDeviceManagementService.

542 **8.10.1.1 CIM_SharedDeviceManagementService.ElementName Property**

543 When an instance of CIM_EnabledLogicalElementCapabilities is associated with the
 544 CIM_SharedDeviceManagementService instance and the
 545 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE, the
 546 implementation shall allow the ModifyInstance operation to change the value of the ElementName
 547 property of the CIM_SharedDeviceManagementService instance. The ModifyInstance operation shall

548 enforce the length restriction specified in the MaxElementNameLen property of the
549 CIM_EnabledLogicalElementCapabilities.

550 When an instance of CIM_EnabledLogicalElementCapabilities is not associated with the
551 CIM_SharedDeviceManagementService instance, or the ElementNameEditSupported property of the
552 CIM_EnabledLogicalElementCapabilities has a value of FALSE, the implementation shall not allow the
553 ModifyInstance operation to change the value of the ElementName property of the
554 CIM_SharedDeviceManagementService instance.

555 8.11 CIM_SharingDependency

556 Table 13 lists implementation requirements for operations. If implemented, these operations shall be
557 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 13, all operations
558 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

559 NOTE: Related profiles may define additional requirements on operations for the profile class.

560 **Table 13 – CIM_SharingDependency Operations**

Operation	Requirement	Messages
ModifyInstance	Optional. See section 8.11.1.	None
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

561 8.11.1 ModifyInstance

562 When the ModifyInstance operation is supported for an instance of CIM_SharingDependency, the
563 ModifyInstance operation shall not modify the CurrentAccess property.

564 8.12 CIM_SystemDevice

565 Table 14 lists implementation requirements for operations. If implemented, these operations shall be
566 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 14, all operations
567 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

568 NOTE: Related profiles may define additional requirements on operations for the profile class.

569 **Table 14 – CIM_SystemDevice Operations**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

570 9 Use Cases

571 This section outlines the use cases specific to managing shared devices. Use cases are informative and
572 not intended to define the requirements for conformance.

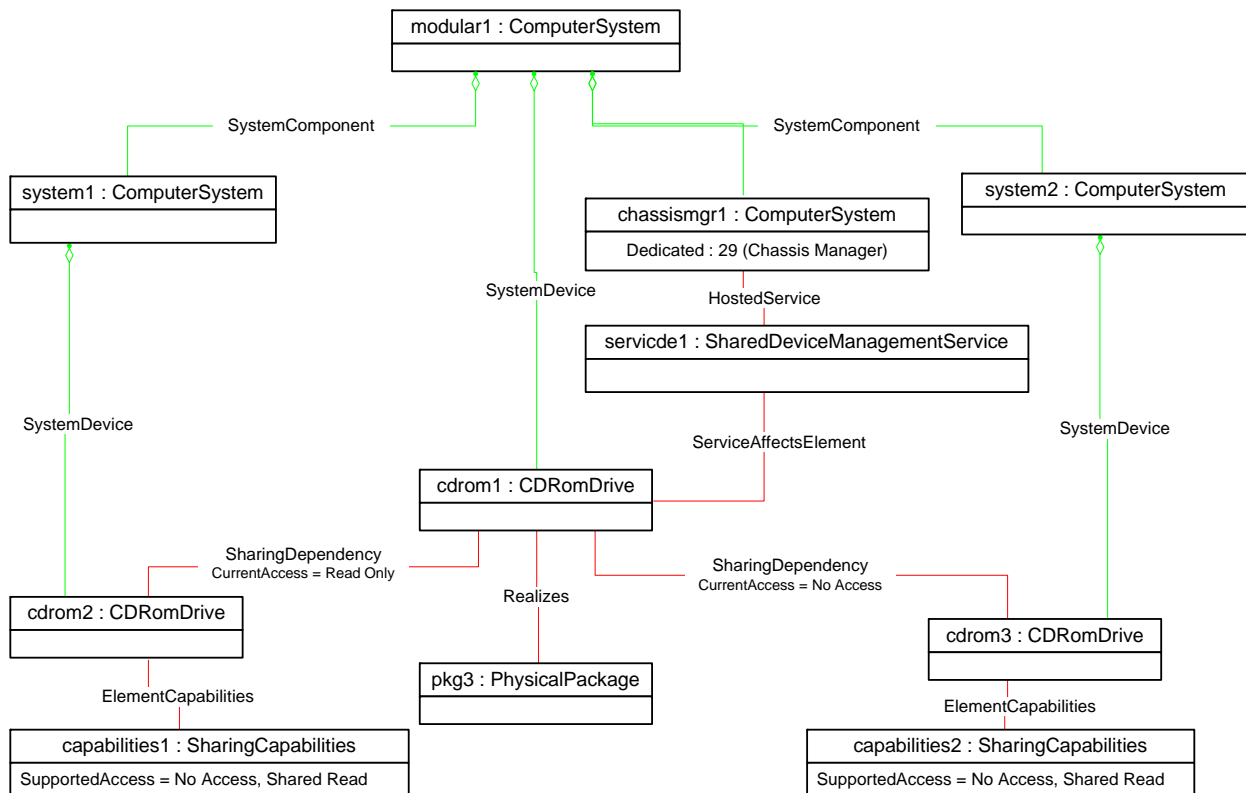
573 **9.1 Object Diagrams**

574 Figure 2 and Figure 3 represent specific possible instantiations and may be referenced by the described
 575 use cases.

576 Figure 2 illustrates the set of classes used to model the management of shared devices. A single instance
 577 of CIM_ComputerSystem (*modular1*) exists to model the modular enclosure. An instance of
 578 CIM_ComputerSystem exists to model each blade server, as well as the chassis manager.

579 A single instance of CIM_CDROMDrive (*cdrom1*) exists to model the shared CD-ROM drive. The CD-
 580 ROM drive belongs to the modular system, so this instance is associated with *modular1*. An additional
 581 instance of CIM_CDROMDrive (*cdrom2* and *cdrom3*) exists for each ComputerSystem instance with
 582 which the CD-ROM drive can be shared. Each instance is associated with exactly one ComputerSystem
 583 instance through the CIM_SystemDevice association. *cdrom1* is a Real Logical Device. *cdrom2* and
 584 *cdrom3* are Sharing Logical Devices.

585 Each instance is also associated with the underlying CIM_CDROMDrive instance through the
 586 CIM_SharingDependency association. Each CIM_CDROMDrive instance that represents a sharing
 587 ComputerSystem's view of the device is associated with an instance of CIM_SharingCapabilities. This
 588 represents the ability of the CIM_ComputerSystem to use the shared device. For example, *capabilities1*
 589 represents the ability of the blade server represented by *system1* to use the shared CDROM drive of the
 590 modular system. In the implementation being modeled, management of the shared devices is performed
 591 by the chassis management module. An instance of CIM_SharedDeviceManagementService (*service1*)
 592 exists and is associated with the chassis manager through an instance of CIM_HostedService. The
 593 CIM_ServiceAffectsElement association indicates that the CIM_SharedDeviceManagementService
 594 manages the shared CD-ROM drive.

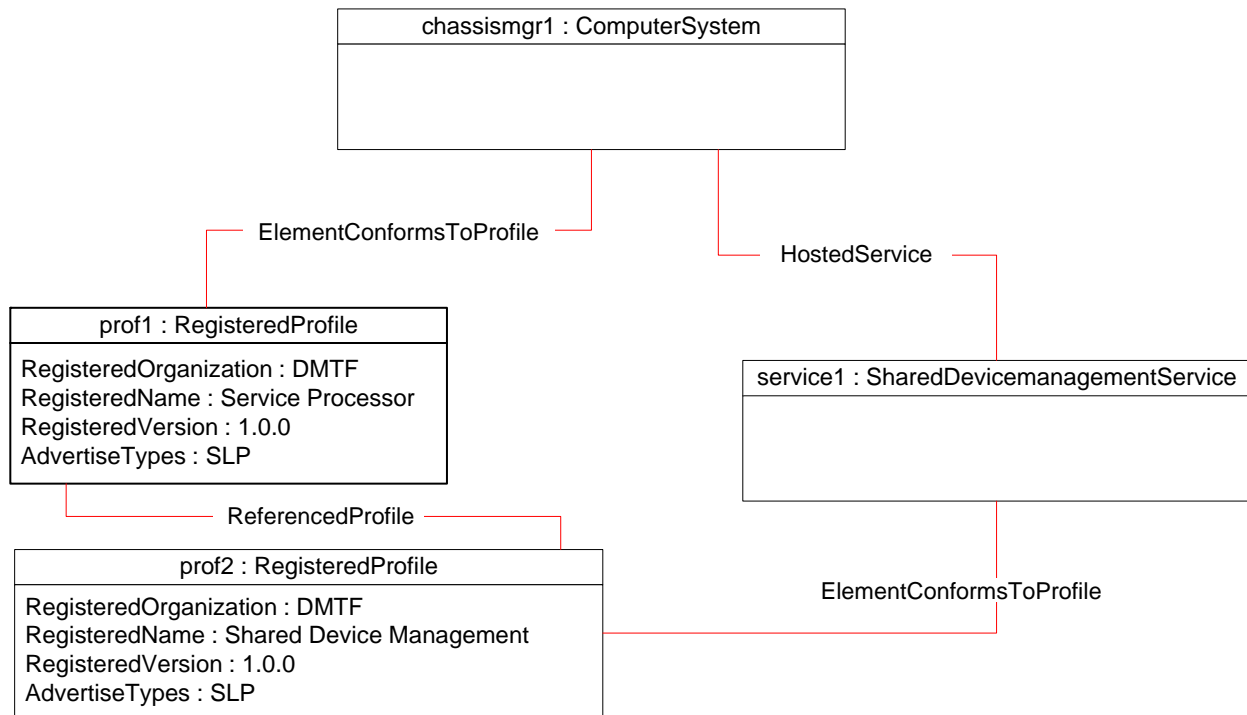


595

596

Figure 2 – Shared Device Management

597 Figure 3 is an object diagram that indicates how an implementation would advertise conformance with the
 598 *Shared Device Management Profile*. *prof2* is an instance of *CIM_RegisteredProfile* that advertises the
 599 *Shared Device Management Profile 1.0.0*. It is associated with the compliant instance of
 600 *CIM_SharedDeviceManagementService* through an instance of *CIM_ElementConformsToProfile*.



601

602

Figure 3 – Registered Profile

603 9.2 Determining If a Logical Device Is a Shared Device

604 A client can determine whether an instance of *CIM_LogicalDevice* corresponds to a shared component as
 605 follows:

- 606 • If the *CIM_LogicalDevice* instance is the *PartComponent* reference in an instance of the
 607 *CIM_SharingDependency* association, the *CIM_LogicalDevice* instance is a *Sharing Logical*
 608 *Device*.
- 609 • If the *CIM_LogicalDevice* instance is not the *PartComponent* reference in an instance of the
 610 *CIM_SharingDependency* association, the *CIM_LogicalDevice* instance is not a *Sharing Logical*
 611 *Device*. (However, it may not correspond to a real hardware resource because it could be a
 612 virtual or pseudo device as defined by a different profile.)

613 9.3 Finding the *CIM_LogicalDevice* Instance for a Real Device

614 A client can determine which *CIM_LogicalDevice* instance is the *Real Logical Device* instance as follows:

- 615 • The *CIM_LogicalDevice* instance is never the *PartComponent* reference in an instance of the
 616 *CIM_SharingDependency* association.
- 617 • The *CIM_LogicalDevice* instance is associated with an instance of *CIM_PhysicalElement*
 618 through an instance of *CIM_Realizes*.
- 619 • The *CIM_LogicalDevice* instance is associated with an instance of
 620 *CIM_SharedDeviceManagementService* through an instance of *CIM_ServiceAffectsElement*.

621 **9.4 Determining How a Device Can Be Shared**

622 Clients should be able to determine how a shared device can be shared within the chassis:

- 623 • Concurrent read
- 624 • Single owner
- 625 • Concurrent write
- 626 • Unavailable
- 627 • Access restricted

628 A client can determine how a device can be shared in the enclosure as follows:

- 629 1) Find the Real Logical Device that represents the shared device by using the directions in
630 section 9.2.
- 631 2) For each instance of CIM_SharingDependency that references this CIM_LogicalDevice
632 instance, find the CIM_LogicalDevice instance that is the Dependent reference.
- 633 3) Find the CIM_ElementCapabilities association instances that reference this second
634 CIM_LogicalDevice instance.
- 635 4) Use the CIM_ElementCapabilities association to find a referenced instance of
636 CIM_DeviceSharingCapabilities.

637 Each instance of CIM_DeviceSharingCapabilities indicates the ability of a system to use the shared
638 device. Thus, the types of sharing available for the device are defined as the union of the types of sharing
639 indicated by each instance of CIM_DeviceSharingCapabilities.

640 **9.5 Determining System Access to a Shared Device**

641 Clients can determine the ability of a system to use a shared device as follows:

- 642 1) Find the Real Logical Device that represents the shared device by using the directions in
643 section 9.2.
- 644 2) For each instance of CIM_SharingDependency that references this CIM_LogicalDevice
645 instance, find the CIM_LogicalDevice instance that is the Dependent reference (a Sharing
646 Logical Device).
- 647 3) Determine if the CIM_LogicalDevice instance is associated through an instance of
648 CIM_SystemDevice with the instance of CIM_ComputerSystem that represents the system that
649 is the subject of the query.
- 650 4) If a Sharing Logical Device associated with the subject CIM_ComputerSystem instance is not
651 found, the system cannot use the shared device.
- 652 5) If a Sharing Logical Device instance is found, find the CIM_ElementCapabilities association
653 instances that reference this second CIM_LogicalDevice instance.
- 654 6) Use the CIM_ElementCapabilities association to find a referenced instance of
655 CIM_DeviceSharingCapabilities.
- 656 7) Query the SupportedAccessModes property of the CIM_DeviceSharingCapabilities instance to
657 determine the utilization of the shared device supported for the subject system.

658 9.6 Changing Shared Device Access

659 Clients can modify access of systems within the chassis to a shared device as follows:

- 660 1) Find the Sharing System that represents the system whose access the client wants to modify.
- 661 2) Find the Real Logical Device that represents the shared device for which the client wants to
662 modify the utilization of the system.
- 663 3) Find the instances of the CIM_ServiceAffectsElement association that reference the Real
664 Logical Device.
 - 665 – For each instance of CIM_ServiceAffectsElement, the client will look at the
666 ServiceProvided reference to find an instance of CIM_SharedDeviceManagementService.
- 667 4) When the client has found the instance of CIM_SharedDeviceManagementService, it will invoke
668 the ShareDevice() method on the instance, specifying the Sharing System, the Real Logical
669 Device, the desired access, whether to force access to be granted, and how long the client
670 wants to wait for access to be granted.

671 9.7 Determining If ElementName Can Be Modified

672 For a given instance of CIM_SharedDeviceManagementService, a client can determine whether it can
673 modify the ElementName as follows:

- 674 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the target
675 instance.
- 676 2) Query the value of the ElementNameEditSupported property of the
677 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
678 the ElementName property of the target instance.

679 10 CIM Elements

680 Table 15 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
681 implemented as described in Table 15. Sections 7 (“Implementation Requirements”) and 8 (“Methods”)
682 may impose additional requirements on these elements.

683 **Table 15 – Required CIM Elements: Shared Device Management Profile**

Element Name	Requirement	Notes
Classes		
CIM_DeviceSharingCapabilities	Mandatory	See section 10.1.
CIM_ElementCapabilities	Mandatory	See sections 10.2 and 10.3.
CIM_EnabledLogicalElementCapabilities	Mandatory	See section 10.4.
CIM_HostedService	Mandatory	See section 10.5.
CIM_LogicalDevice	Mandatory	See section 10.6.
CIM_RegisteredProfile	Mandatory	See section 10.7.
CIM_ServiceAffectsElement	Mandatory	See section 10.8.
CIM_SharedDeviceManagementService	Mandatory	See section 10.9.
CIM_SharingDependency	Mandatory	See section 10.10.
CIM_SystemDevice	Mandatory	See section 10.11.
Indications		
None defined in this profile		

684 **10.1 CIM_DeviceSharingCapabilities**

685 CIM_DeviceSharingCapabilities indicates the ability of a client system to use a shared device.

686 **Table 16 – Class: CIM_DeviceSharingCapabilities**

Properties	Requirement	Notes
SupportedAccessModes	Mandatory	See section 7.1.4.2.
InstanceID	Mandatory	None
ElementName	Mandatory	pattern ".*"

687 **10.2 CIM_ElementCapabilities – SharingCapabilities**

688 CIM_ElementCapabilities associates an instance of CIM_DeviceSharingCapabilities with the
689 CIM_LogicalDevice instance that represents the Sharing Logical Device.

690 **Table 17 – Class: CIM_ElementCapabilities – SharingCapabilities**

Properties	Requirement	Notes
ManagedElement	Mandatory	See section 7.1.2.1. Cardinality 1..*
Capabilities	Mandatory	See section 7.1.2.2. Cardinality 1..*

691 **10.3 CIM_ElementCapabilities – EnabledLogicalElementCapabilities**

692 CIM_ElementCapabilities associates an instance of CIM_EnabledLogicalElementCapabilities with the
693 Central Instance.

694 **Table 18 – Class: CIM_ElementCapabilities – EnabledLogicalElementCapabilities**

Properties	Requirement	Notes
ManagedElement	Mandatory	Cardinality 1..* This property shall be a reference to the Central Instance.
Capabilities	Mandatory	Cardinality 1 This property shall be a reference to an instance of CIM_EnabledLogicalElementCapabilities.

695 **10.4 CIM_EnabledLogicalElementCapabilities**

696 CIM_EnabledLogicalElementCapabilities indicates support for managing the state of the service.

697 **Table 19 – Class: CIM_EnabledLogicalElementCapabilities**

Properties	Requirement	Notes
InstanceID	Mandatory	None
RequestedStatesSupported	Mandatory	See sections 7.2.1.1.1 and 7.2.2.1.1.
ElementNameEditSupported	Mandatory	See sections 7.2.3.1.1 and 7.2.4.1.1.
MaxElementNameLen	Conditional	See sections 7.2.3.1.2 and 7.2.4.1.2.
ElementName	Mandatory	pattern ".*"

698 **10.5 CIM_HostedService**

699 CIM_HostedService associates an instance of CIM_SharedDeviceManagementService with the
700 CIM_ComputerSystem to which it is scoped.

701 **Table 20 – Class: CIM_HostedService**

Properties	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	This property shall be a reference to the Central Instance. Cardinality 1..*

702 **10.6 CIM_LogicalDevice**

703 Implementations of this profile will create an instance of the appropriate subclass of CIM_LogicalDevice
704 to represent the Sharing Logical Device.

705 **Table 21 – Class: CIM_LogicalDevice**

Properties	Requirement	Notes
SystemCreationClassName	Mandatory	None
SystemName	Mandatory	None
CreationClassName	Mandatory	None
Name	Mandatory	None
ElementName	Mandatory	pattern ".*"

706 **10.7 CIM_RegisteredProfile**

707 CIM_RegisteredProfile identifies the *Shared Device Management Profile* so that a client can determine
708 whether an instance of CIM_ComputerSystem is conformant with this profile. The CIM_RegisteredProfile
709 class is defined by the [Profile Registration Profile](#). With the exception of the mandatory values specified
710 for the properties in Table 22, the behavior of the CIM_RegisteredProfile instance is in accordance with
711 the constraints specified in the [Profile Registration Profile](#).

712 **Table 22 – Class: CIM_RegisteredProfile**

Properties	Requirement	Notes
RegisteredName	Mandatory	This property shall have a value of "Shared Device Management".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.0".
OwningEntity	Mandatory	This property shall have a value of "DMTF".

713 NOTE: Previous versions of this document included the suffix "Profile" for the RegisteredName value. If
714 implementations querying for the RegisteredName value find the suffix "Profile", they should ignore the suffix, with
715 any surrounding white spaces, before any comparison is done with the value as specified in this document.

716 **10.8 CIM_ServiceAffectsElement**

717 CIM_ServiceAffectsElement associates an instance of CIM_SharedDeviceManagementService with an
 718 instance of CIM_LogicalDevice that represents a shared logical device that the service can manage.

719 **Table 23 – Class: CIM_ServiceAffectsElement**

Properties	Requirement	Notes
ServiceProvided	Mandatory	This property shall be a reference to the Central Instance of the profile. Cardinality 1..*
UserOfService	Mandatory	See section 7.1.3.2. Cardinality 1..*
ElementAffects	Mandatory	See section 7.1.3.1.

720 **10.9 CIM_SharedDeviceManagementService**

721 CIM_SharedDeviceManagementService represents the ability to control access to a shared device.

722 **Table 24 – Class: CIM_SharedDeviceManagementService**

Properties	Requirement	Notes
SystemCreationClassName	Mandatory	None
SystemName	Mandatory	None
CreationClassName	Mandatory	None
Name	Mandatory	None
EnabledState	Mandatory	See sections 7.2.2.3 and 7.2.1.3.
RequestedState	Mandatory	See sections 7.2.1.2 and 7.2.2.2.
RequestStateChange()	Conditional	See section 8.2.
ShareDevice()	Mandatory	See section 8.1.
ElementName	Mandatory	pattern ".*" See sections 7.2.3 and 7.2.4.
OperationalStatus	Mandatory	None
HealthState	Mandatory	None

723 **10.10 CIM_SharingDependency**

724 CIM_SharingDependency associates an instance of CIM_SharedDeviceManagementService with an
 725 instance of CIM_LogicalDevice that represents a shared logical device that the service can manage.

726 **Table 25 – Class: CIM_SharingDependency**

Properties	Requirement	Notes
Antecedent	Mandatory	Reference. See section 7.1.4.2. Cardinality 1
Dependent	Mandatory	Reference. See section 7.1.4.1. Cardinality 1..*
Current Access	Mandatory	This property shall identify the current system access to the shared device, which is the Antecedent reference.

727 **10.11 CIM_SystemDevice**

728 CIM_SystemDevice associates a Sharing Logical Device with an instance of CIM_ComputerSystem.

729 **Table 26 – Class: CIM_SystemDevice**

Properties	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to an instance of CIM_ComputerSystem. Cardinality 1
Dependent	Mandatory	This property shall be a reference to a Sharing Logical Device. Cardinality 1

730

731
732
733
734

ANNEX A
(informative)
Change Log

Version	Date	Description
1.0.0	6/16/2009	DMTF Standard Release

735
736