



1

2

3

4

Document Number: DSP1058

Date: 2010-09-15

Version: 1.0.1

5 **Base Desktop and Mobile Profile**

6 **Document Type: Specification**

7 **Document Status: DMTF Standard**

8 **Document Language: en-US**

9 Copyright Notice

10 Copyright © 2010 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25 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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	3 Terms and Definitions	8
37	4 Symbols and Abbreviated Terms	9
38	5 Synopsis.....	9
39	6 Description	10
40	6.1 Representation of System Power State.....	11
41	7 Implementation Requirements	12
42	7.1 Base Desktop and Mobile System.....	12
43	7.2 Management of Base Desktop Mobile Components	13
44	7.3 State Management.....	13
45	7.4 Simple Identity Management	17
46	7.5 Role Based Authorization	17
47	8 Methods.....	17
48	9 Use Cases.....	17
49	9.1 Object Diagrams	17
50	9.2 Determine the System Model and Serial Number	23
51	9.3 Power On a System	24
52	9.4 Power Off a System	24
53	9.5 Shut Down and Restart a System.....	24
54	9.6 Perform System Power Control	24
55	9.7 Determining the System Power State.....	24
56	9.8 Determine the Number of Processors in the System	25
57	9.9 Determine the Number of Processors That the System Can Hold.....	25
58	10 CIM Elements.....	25
59	10.1 CIM_ComputerSystem.....	26
60	10.2 CIM_ComputerSystemPackage	26
61	10.3 CIM_EnabledLogicalElementCapabilities.....	26
62	10.4 CIM_PhysicalPackage	26
63	10.5 CIM_RegisteredProfile.....	26
64	ANNEX A (Informative) Change Log	28
65		

66 Figures

67	Figure 1 – Base Desktop and Mobile Profile: Class Diagram.....	11
68	Figure 2 – Profile Registration.....	18
69	Figure 3 – Power Management and Time Service	19
70	Figure 4 – Advanced Power Management.....	20
71	Figure 5 – Logical Topology.....	21
72	Figure 6 – Physical Topology.....	22
73	Figure 7 – Logical to Physical Mapping	23
74		

75 **Tables**

76	Table 1 – Referenced Profiles	10
77	Table 2 – EnabledState and ACPI State Equivalence	13
78	Table 3 – PowerState and EnabledState Values (Required Equivalence)	15
79	Table 4 – PowerState and EnabledState Values (Recommended Equivalence)	15
80	Table 6 – RequestedPowerState and RequestedState Values (Recommended Equivalence)	16
81	Table 8 – PowerStatesSupported and RequestedStatesSupported Values (Recommended Equivalence)	17
82	Table 9 – CIM Elements: Base Desktop and Mobile Profile	25
83	Table 10 – Class: CIM_ComputerSystem	26
84	Table 11 – Class: CIM_ComputerSystemPackage	26
85	Table 12 – Class: CIM_EnabledLogicalElementCapabilities	26
86	Table 13 – Class: CIM_RegisteredProfile	27

87

88

Foreword

89 The *Base Desktop and Mobile Profile* (DSP1058) was prepared by the Desktop Mobile Working Group
90 and Physical Platform Profiles Working Group of the DMTF.

91 DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems
92 management and interoperability. For information about the DMTF, see <http://www.dmtf.org>.

93 **Acknowledgments**

94 The authors wish to acknowledge the following people, including all the editors and contributors of the
95 *Base Server Profile*, on which this profile is based.

96 Editors:

- 97 • Hemal Shah – Broadcom
- 98 • Jianwen Yin – Dell

99 Contributors:

- 100 • Hemal Shah – Broadcom
- 101 • Jon Hass – Dell
- 102 • Khachatur Papanyan – Dell
- 103 • Jeff Hilland – HP
- 104 • Christina Shaw – HP
- 105 • Aaron Merkin – IBM
- 106 • Perry Vincent – Intel
- 107 • John Leung – Intel
- 108 • Jianwen Yin – Dell

109

110

111

Introduction

112 The information in this specification should be sufficient for a provider or consumer of this data to
113 unambiguously identify the classes, properties, methods, and values that shall be instantiated and
114 manipulated to represent and manage a monolithic desktop or mobile computer and its subsystems using
115 the DMTF Common Information Model (CIM) core and extended model definitions.

116 The target audience for this specification is implementers who are writing CIM-based providers or
117 consumers of management interfaces that represent the components described in this document.

118

Base Desktop and Mobile Profile

119 1 Scope

120 The *Base Desktop and Mobile Profile* is an autonomous profile that defines the classes used to describe
121 monolithic desktop or mobile computer hardware and related software. The scope of this profile is limited
122 to monolithic desktop or mobile computer hardware and related software that are directly realized in
123 physical components. The profiles referenced by the *Base Desktop and Mobile Profile* extend the
124 management capabilities described in this profile.

125 2 Normative References

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

129 *Advanced Configuration and Power Interface Specification*, revision 3.0,
130 <http://www.acpi.info/Downloads/ACPIspec30.pdf>

131 DMTF DSP0004, *CIM Infrastructure Specification 2.6*,
132 http://www.dmtf.org/standards/published_documents/DSP0004_2.6.pdf

133 DMTF DSP0200, *CIM Operations over HTTP 1.3*,
134 http://www.dmtf.org/standards/published_documents/DSP0200_1.3.pdf

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

137 DMTF DSP1009, *Sensors Profile 1.0*,
138 http://www.dmtf.org/standards/published_documents/DSP1009_1.0.pdf

139 DMTF DSP1011, *Physical Asset Profile 1.0*,
140 http://www.dmtf.org/standards/published_documents/DSP1011_1.0.pdf

141 DMTF DSP1012, *Boot Control Profile 1.0*,
142 http://www.dmtf.org/standards/published_documents/DSP1012_1.0.pdf

143 DMTF DSP1013, *Fan Profile 1.0*,
144 http://www.dmtf.org/standards/published_documents/DSP1013_1.0.pdf

145 DMTF DSP1015, *Power Supply Profile 1.0*,
146 http://www.dmtf.org/standards/published_documents/DSP1015_1.0.pdf

147 DMTF DSP1022, *CPU Profile 1.0*,
148 http://www.dmtf.org/standards/published_documents/DSP1022_1.0.pdf

149 DMTF DSP1023, *Role Based Authorization Profile 1.0*,
150 http://www.dmtf.org/standards/published_documents/DSP1023_1.0.pdf

151 DMTF DSP1026, *System Memory Profile 1.0*,
152 http://www.dmtf.org/standards/published_documents/DSP1026_1.0.pdf

153 DMTF DSP1027, *Power State Management Profile 1.0*,
154 http://www.dmtf.org/standards/published_documents/DSP1027_1.0.pdf

- 155 DMTF DSP1033, *Profile Registration Profile 1.0*,
156 http://www.dmtf.org/standards/published_documents/DSP1033_1.0.pdf
- 157 DMTF DSP1034, *Simple Identity Management Profile 1.0*,
158 http://www.dmtf.org/standards/published_documents/DSP1034_1.0.pdf
- 159 DMTF DSP1052, *Computer System Profile 1.0*,
160 http://www.dmtf.org/standards/published_documents/DSP1052_1.0.pdf
- 161 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
162 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>

163 3 Terms and Definitions

164 For the purposes of this document, the following terms and definitions apply. For the purposes of this
165 document, the terms and definitions given in [DSP1033](#) and [DSP1001](#) also apply.

166 3.1

167 **can**

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

169 3.2

170 **cannot**

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

172 3.3

173 **conditional**

174 indicates requirements to be followed strictly to conform to the document when the specified conditions
175 are met

176 3.4

177 **mandatory**

178 indicates requirements to be followed strictly to conform to the document and from which no deviation is
179 permitted

180 3.5

181 **may**

182 indicates a course of action permissible within the limits of the document

183 3.6

184 **need not**

185 indicates a course of action permissible within the limits of the document

186 3.7

187 **optional**

188 indicates a course of action permissible within the limits of the document

189 3.8

190 **referencing profile**

191 indicates a profile that owns the definition of this class and can include a reference to this profile in its
192 "Referenced Profiles" table

193 3.9

194 **shall**

195 indicates requirements to be followed strictly to conform to the document and from which no deviation is
196 permitted

- 197 **3.10**
198 **shall not**
199 indicates requirements to be followed strictly to conform to the document and from which no deviation is
200 permitted
- 201 **3.11**
202 **should**
203 indicates that among several possibilities, one is recommended as particularly suitable, without
204 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 205 **3.12**
206 **should not**
207 indicates that a certain possibility or course of action is deprecated but not prohibited
- 208 **3.13**
209 **unspecified**
210 indicates that this profile does not define any constraints for the referenced CIM element or operation

211 **4 Symbols and Abbreviated Terms**

- 212 **4.1**
213 **ACPI**
214 Advanced Configuration and Power Interface specification

215 **5 Synopsis**

- 216 **Profile Name:** Base Desktop and Mobile
- 217 **Version:** 1.0.1
- 218 **Organization:** DMTF
- 219 **CIM schema version:** 2.20
- 220 **Specializes:** DMTF *Computer System Profile* 1.0.0 ([DSP1052](#))
- 221 **Central Class:** CIM_ComputerSystem
- 222 **Scoping Class:** CIM_ComputerSystem
- 223 The *Base Desktop and Mobile Profile* is an autonomous profile that provides the capability to manage
224 monolithic desktop or mobile computer hardware and related software.
- 225 The Central Class of the *Base Desktop and Mobile Profile* shall be CIM_ComputerSystem. The Central
226 Instance shall be an instance of CIM_ComputerSystem. The Scoping Class shall be
227 CIM_ComputerSystem. The Scoping Instance shall be the Central Instance. Table 1 lists profiles upon
228 which this profile has a dependency.
- 229 Note that the behavioral definitions for many of the profiles are inherited from the abstract [Computer](#)
230 [System Profile](#). Therefore, they are not referenced in Table 1. Examples are the [System Memory Profile](#)
231 and the [Sensors Profile](#).

232

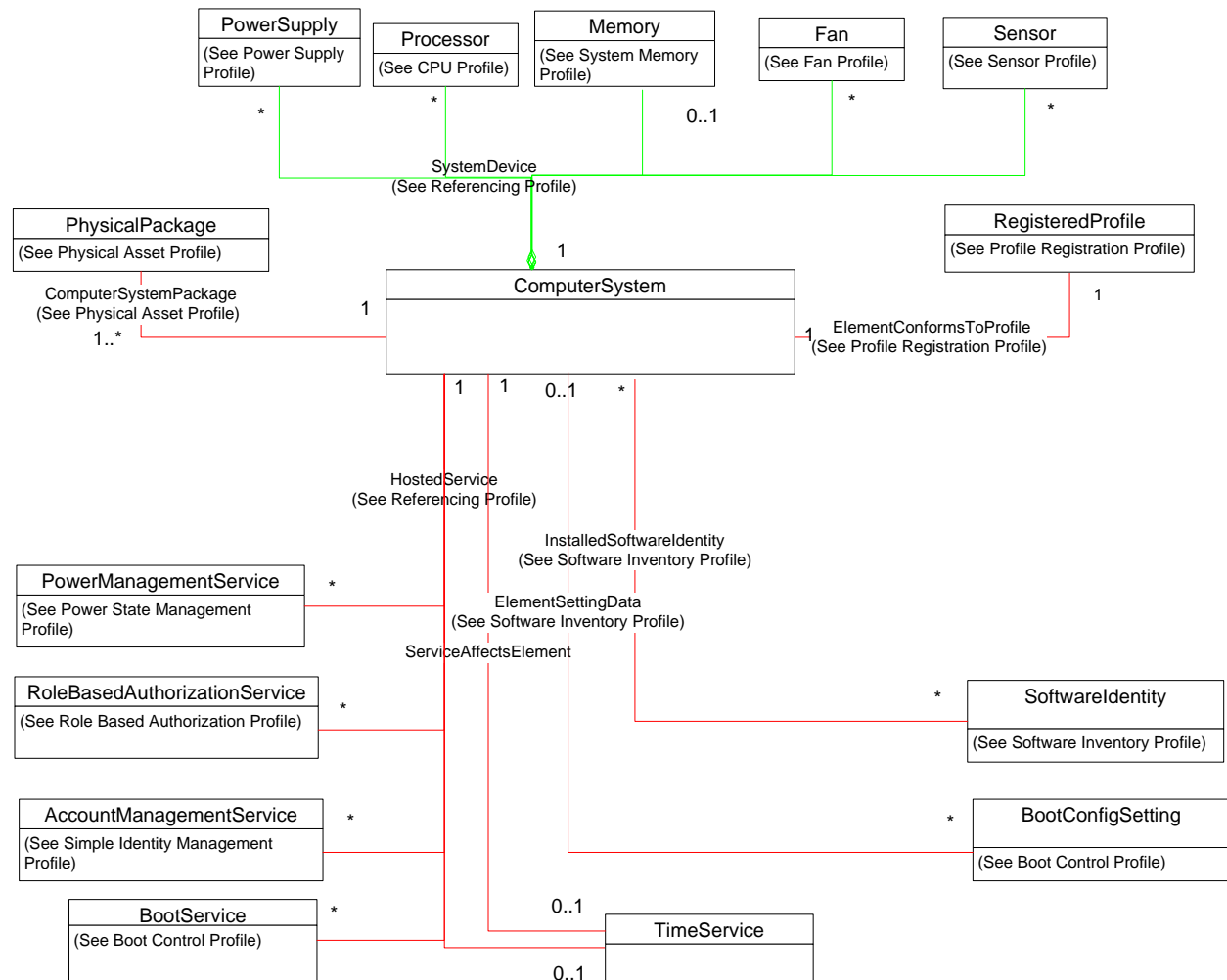
Table 1 – Referenced Profiles

Profile Name	Organization	Version	Relationship	Behavior
Computer System	DMTF	1.0	Specializes	None
Fan	DMTF	1.0	Optional	See 7.2.1.
Physical Asset	DMTF	1.0	Mandatory	See 7.1.2.
Power State Management	DMTF	1.0	Optional	See 7.3.2.
Power Supply	DMTF	1.0	Optional	See 7.2.2.
Profile Registration	DMTF	1.0	Mandatory	None
Simple Identity Management	DMTF	1.0	Optional	See 7.4.
Role Based Authorization	DMTF	1.0	Optional	See 7.5.

233 6 Description

234 The *Base Desktop and Mobile Profile* is an autonomous profile that defines the minimum top-level object
 235 model needed to model monolithic desktop or mobile computer hardware and related software. Other
 236 profiles add additional management objects to this basic desktop mobile model to provide system
 237 configuration, boot control, and other provisioning capabilities. CIM_ComputerSystem represents the
 238 desktop mobile system. CIM_TimeService provides the ability to manage the system time.

239 Figure 1 presents the class schema for the *Base Desktop and Mobile Profile*. For simplicity, the prefix
 240 *CIM_* has been removed from the names of the classes.



241

242

Figure 1 – Base Desktop and Mobile Profile: Class Diagram

243 Note that the behavioral constraints for many of the profiles identified in Figure 1 are inherited from the
 244 specialized [Computer System Profile](#). Therefore, although they are shown in Figure 1, they are not
 245 referenced in this specification. Examples are the [System Memory Profile](#) and the [Sensors Profile](#).

246 **6.1 Representation of System Power State**

247 Normative requirements for the representation of the power state of a system are expressed in 7.3. The
 248 following informative text provides background on the approach taken to modeling the power state of a
 249 system.

250 The *Base Desktop and Mobile Profile* identifies two complementary approaches to representing the
 251 power state of a base desktop and mobile system: simple on/off management through the
 252 RequestedState and EnabledState properties, and the RequestStateChange() method. Definitions for the
 253 2 (Enabled) and 3 (Disabled) values for EnabledState use industry-standard ACPI definitions. Alternately,
 254 if an implementation wants to support more granular or complex power management behavior; the [Power](#)
 255 [State Management Profile](#) can be implemented.

256 The power management behavior and system power states specified in the [Power State Management](#)
 257 [Profile](#) are a superset of the function and states that are represented using the EnabledState and
 258 RequestedState properties of CIM_ComputerSystem. That is, the EnabledState and RequestedState
 259 properties are sufficient to represent ACPI states S0 and S5. Implementing the [Power State Management](#)

260 [Profile](#) provides the ability to represent additional ACPI states. Although some of the values of
261 EnabledState and PowerState are equivalent, this equivalency results from them being mapped to
262 identical ACPI states rather than being defined in terms of each other. With this method, for the subset of
263 values for EnabledState and RequestedState for which ACPI states are defined, there is a one-to-one
264 correspondence with a legal value for the PowerState and RequestedPowerState properties.

265 The method of defining the states that are expressible through the [Power State Management Profile](#) as a
266 superset of those possible with EnabledState and RequestedState is contrasted with the discarded
267 alternative method of using the implementation of the [Power State Management Profile](#) to provide a
268 refinement of the interpretation of the EnabledState and RequestedState values. If this latter, discarded
269 method had been used, multiple values of PowerState and RequestedPowerState would have been
270 mapped onto the less granular values for the EnabledState and RequestedState properties.

271 7 Implementation Requirements

272 The *Base Desktop and Mobile Profile* consists of definitions for CIM_ComputerSystem,
273 CIM_PhysicalPackage, CIM_TimeService, and their related EnabledLogicalElementCapabilities. Other
274 related subsystem classes such as CIM_LogicalDevice, CIM_Collection, and CIM_RecordLog are defined
275 in their respective profiles.

276 Requirements for propagating and formulating certain properties of the *Base Desktop and Mobile Profile*
277 classes are discussed in this clause. The *Base Desktop and Mobile Profile* defines how to model the
278 system's logical aspects, and the [Physical Asset Profile](#) defines how to model the system's physical
279 aspects.

280 The list of all methods can be found in clause 8 ("Methods"), and the list of properties can be found in
281 clause 10 ("CIM Elements").

282 7.1 Base Desktop and Mobile System

283 There shall be an instance of CIM_ComputerSystem to represent the system being modeled.

284 7.1.1 Identifying a Base Desktop Mobile

285 This clause details constraints beyond those specified in the [Computer System Profile](#) for using the
286 IdentifyingDescriptions and OtherIdentifyingInfo properties to identify a computer system.

287 7.1.1.1 CIM:GUID

288 For each unique value of the PlatformGUID property of an instance of CIM_ComputerSystemPackage
289 that references the CIM_ComputerSystem instance, the IdentifyingDescriptions property shall contain the
290 value "CIM:GUID", and the corresponding array position of the OtherIdentifyingInfo property shall contain
291 the value of the PlatformGUID property.

292 7.1.1.2 CIM:Model:SerialNumber

293 For each unique combination of the values of the ModelNumber and SerialNumber properties of an
294 associated instance of CIM_PhysicalPackage, the IdentifyingDescriptions property of
295 CIM_ComputerSystem instance shall contain the value " CIM:Model:SerialNumber ". The value of the
296 corresponding OtherIdentifyInfo array index shall be of the form specified in the [Computer System Profile](#).
297 The <Model Number> portion of this value shall match the value of the Model property of the
298 CIM_PhysicalPackage instance. The <Serial Number> portion of this value shall match the value of the
299 SerialNumber property of the same CIM_PhysicalPackage instance.

300 **7.1.1.3 CIM:Tag**

301 For each unique value of the Tag property of an associated instance of CIM_PhysicalPackage, the
 302 IdentifyingDescriptions property of the CIM_ComputerSystem instance shall contain the value "CIM:Tag",
 303 and the corresponding array position of the OtherIdentifyingInfo property shall contain the value of the
 304 Tag property of the CIM_PhysicalPackage instance.

305 **7.1.2 Representing the Physical Packaging**

306 The physical packaging for a system shall be modeled in a way that is compliant with the requirements
 307 specified in the [Physical Asset Profile](#). At least one instance of CIM_PhysicalPackage shall be associated
 308 with the Central Instance through the CIM_ComputerSystemPackage association.

309 **7.2 Management of Base Desktop Mobile Components**

310 The following subclauses detail the requirements for management of components of the system in
 311 addition to those specified in the [Computer System Profile](#).

312 **7.2.1 Instrumentation of Fans**

313 A system can contain one or more fans that provide cooling for the system. If the fans of the system are
 314 instrumented, the instrumentation should be conformant with the [Fan Profile](#). If the fans of the system are
 315 instrumented in conformance with the [Fan Profile](#), and the Central Instance of this profile shall be
 316 associated with one of more instances of the Central Class of the [Fan Profile](#) through the
 317 CIM_SystemDevice association.

318 **7.2.2 Instrumentation of Power Supplies**

319 A system can contain one or more power supplies that provide power to the system. If the power supplies
 320 of the system are instrumented, the instrumentation should be conformant with the [Power Supply Profile](#).
 321 If the power supplies of the system are instrumented in conformance with the [Power Supply Profile](#), the
 322 Central Instance of this profile shall be associated with one or more instances of the Central Class of the
 323 [Power Supply Profile](#) through the CIM_SystemDevice association.

324 **7.3 State Management**

325 This clause details further constraints related to state management beyond those specified in the
 326 [Computer System Profile](#).

327 **7.3.1 Correspondence of System States and ACPI States**

328 The EnabledState property of CIM_ComputerSystem is defined in terms of ACPI values in order to
 329 provide meaningful context for the interpretation of values for a computer system realized in hardware.
 330 The mappings specified in Table 2 shall be used. Note that the underlying modeled system does not need
 331 to support the ACPI specification.

332 **Table 2 – EnabledState and ACPI State Equivalence**

EnabledState Value	Corresponding ACPI State
2 (Enabled)	G0 or S0 Working
3 (Disabled)	G2 or S5
9 (Quiesce)	G1, S1, S2, S3, or S4

333 7.3.2 Power State Management

334 When the Scoping Instance of the [Power State Management Profile](#) is the Scoping Instance of this
335 profile, the requirements defined in this clause are applicable.

336 7.3.2.1 Power Management Available to System

337 Management of the power state of the system may be supported for the system. When the management
338 of the power state is supported, the [Power State Management Profile](#) shall be implemented and the
339 Central Instance of this profile shall be associated with the Central Instance of the [Power State
340 Management Profile](#) through the CIM_AssociatedPowerManagement association.

341 7.3.2.2 Power Management Hosted on System

342 The system may provide the ability to manage the power state of itself or other systems. When the
343 system provides this ability, the [Power State Management Profile](#) shall be implemented and the Central
344 Instance of this profile shall be associated with the Central Instance of the [Power State Management
345 Profile](#) through the CIM_HostedService association.

346 7.3.3 Relationship between State Management and Power State Management

347 The behavior in this clause is conditional on the implementation of the behavior in 7.3.2.1. When the
348 optional behavior specified in 7.3.2.1 is supported, the state management behavior specified in the
349 [Computer System Profile](#) shall be supported.

350 Power state management may be supported for a system. One reason for supporting power state
351 management is the need to provide more granular management beyond that available through state
352 management. To ensure consistent semantics for state management regardless of whether power state
353 management is supported, constraints on the interaction of power state management and state
354 management must be established when power state management is supported. This clause details these
355 constraints.

356 NOTE: The CIM_ComputerSystem.RequestStateChange() method defined in the [Computer System Profile](#) will
357 cause the values for the CIM_ComputerSystem.EnabledState and CIM_ComputerSystem.RequestedState properties
358 to change. Because of the equivalence requirements stated in the following clauses, this change might result in
359 changes to the values of the CIM_AssociatedPowerManagementService.RequestedPowerState and
360 CIM_AssociatedPowerManagementService.PowerState properties. Likewise, the
361 CIM_PowerManagementService.RequestPowerStateChange() method defined in the [Power State Management
362 Profile](#) will cause the CIM_AssociatedPowerManagementService.RequestedPowerState and
363 CIM_AssociatedPowerManagementService.PowerState properties to change. Because of the equivalence
364 requirements stated in the following clauses, this change might result in changes to the values of the
365 CIM_ComputerSystem.EnabledState and CIM_ComputerSystem.RequestedState properties.

366 7.3.3.1 Relationship between EnabledState and PowerState

367 Table 3 and Table 4 list equivalency requirements for values of the CIM_ComputerSystem.EnabledState
368 property and the CIM_AssociatedPowerManagementService.PowerState property for the instance of
369 CIM_AssociatedPowerManagementService that references the CIM_ComputerSystem instance. When
370 the CIM_AssociatedPowerManagementService.PowerState property has the value listed in the first
371 column of Table 3, the CIM_ComputerSystem.EnabledState property shall have the value listed in the
372 second column. When the CIM_AssociatedPowerManagementService.PowerState property has the value
373 listed in the first column of Table 4, the CIM_ComputerSystem.EnabledState property should have the
374 value listed in the second column. Note that the set of power states that can be represented by the
375 PowerState property is a superset of those power states that are expressible through the EnabledState
376 property alone. Power states expressible through the PowerState property that are not expressible
377 through the EnabledState property are mapped to 5 (Not Applicable).

378

Table 3 – PowerState and EnabledState Values (Required Equivalence)

PowerState Value	Corresponding EnabledState Value
2 (On)	2 (Enabled)
8 (Off – Soft)	3 (Disabled)
12 (Off – Soft Graceful)	3 (Disabled)

379

Table 4 – PowerState and EnabledState Values (Recommended Equivalence)

PowerState Value	Corresponding EnabledState Value
3 (Sleep – Light)	9 (Quiesce)
4 (Sleep – Deep)	9 (Quiesce)
5 (Power Cycle (Off – Soft))	5 (Not Applicable)
6 (Off – Hard)	3 (Disabled)
7 (Hibernate (Off – Soft))	9 (Quiesce)
9 (Power Cycle (Off – Hard))	5 (Not Applicable)
10 (Master Bus Reset)	5 (Not Applicable)
11 (Diagnostic Interrupt (NMI))	5 (Not Applicable)
13 (Off – Hard Graceful)	3 (Disabled)
14 (Master Bus Reset Graceful)	5 (Not Applicable)
15 (Power Cycle (Off – Soft) Graceful)	5 (Not Applicable)
16 (Power Cycle (Off – Hard) Graceful)	5 (Not Applicable)

380

7.3.3.2 Relationship between RequestedState and RequestedPowerState

381
382
383
384
385
386
387
388
389
390
391
392
393

Table 5 and Table 6 list equivalency requirements for values of the CIM_ComputerSystem.RequestedState property and the CIM_AssociatedPowerManagementService.RequestedPowerState property for the instance of CIM_AssociatedPowerManagementService that references the CIM_ComputerSystem instance. When the CIM_AssociatedPowerManagementService.RequestedPowerState property has the value listed in the first column of Table 5, the CIM_ComputerSystem.RequestedState property shall have the value listed in the second column. When the CIM_AssociatedPowerManagementService.RequestedPowerState property has the value listed in the first column of Table 6, the CIM_ComputerSystem.RequestedState property should have the value listed in the second column. Note that the set of power states that can be represented by the RequestedPowerState property is a superset of those power states that are expressible through the RequestedState property alone. Power states expressible through the RequestedPowerState property that are not expressible through the RequestedState property are mapped to 12 (Not Applicable).

394

Table 5 – RequestedPowerState and RequestedState Values (Required Equivalence)

RequestedPowerState Value	Corresponding RequestedState Value
2 (On)	2 (Enabled)
8 (Off – Soft)	3 (Disabled)
13 (Off – Soft Graceful)	3 (Disabled)

395 **Table 6 – RequestedPowerState and RequestedState Values (Recommended Equivalence)**

RequestedPowerState Value	Corresponding RequestedState Value
3 (Sleep – Light)	9 (Quiesce)
4 (Sleep – Deep)	9 (Quiesce)
5 (Power Cycle (Off – Soft))	11 (Reset)
6 (Off – Hard)	3 (Disabled)
7 (Hibernate (Off – Soft))	9 (Quiesce)
9 (Power Cycle (Off – Hard))	12 (Not Applicable)
10 (Master Bus Reset)	11 (Reset)
11 (Diagnostic Interrupt (NMI))	12 (Not Applicable)
12 (Not Applicable)	12 (Not Applicable)
14 (Off – Hard Graceful)	3 (Disabled)
15 (Master Bus Reset Graceful)	11 (Reset)
16 (Power Cycle (Off – Soft) Graceful)	11 (Reset)
17 (Power Cycle (Off – Hard) Graceful)	12 (Not Applicable)

396 **7.3.3.3 Relationship between RequestedStatesSupported and PowerStatesSupported**

397 Table 7 and Table 8 detail equivalency requirements for values of the following properties:

- 398 • the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property for the
399 instance of CIM_EnabledLogicalElementCapabilities that is associated with the
400 CIM_ComputerSystem instance
- 401 • the CIM_PowerManagementCapabilities.PowerStatesSupported property for the instance of
402 CIM_PowerManagementCapabilities that is associated through CIM_ElementCapabilities with the
403 instance of CIM_PowerManagementService that is associated with the CIM_ComputerSystem
404 instance through the CIM_AssociatedPowerManagementService association

405 When the PowerStatesSupported property contains the value listed in the first column of Table 7, the
406 RequestedStatesSupported property shall contain the value listed in the second column. When the
407 PowerStatesSupported property contains the value listed in the first column of Table 8, the
408 RequestedStatesSupported property should contain the value listed in the second column. The
409 RequestedStatesSupported property may contain additional values that correspond to supported states.
410 The PowerStatesSupported property may contain other values; however, corresponding values for
411 RequestedStatesSupported are not defined. The purpose of the PowerStatesSupported property and
412 RequestedStatesSupported property is to indicate the power state changes that can be initiated through
413 the RequestPowerStateChange() method and the RequestStateChange() method, respectively. The
414 absence of a value from the array indicates the absence of support for that power state change. For those
415 power state changes that can be initiated through the RequestPowerStateChange() method but not
416 through the RequestStateChange() method, no mapping is defined because the absence of a value in
417 the RequestedStatesSupported property implicitly indicates a lack of support for initiating the
418 corresponding power state change.

419 **Table 7 – PowerStatesSupported and RequestedStatesSupported Values (Required Equivalence)**

PowerStatesSupported Value	RequestedStatesSupported Value
2 (On)	2 (Enabled)
8 (Off – Soft)	3 (Disabled)
12 (Off – Soft Graceful)	3 (Disabled)

420
421

Table 8 – PowerStatesSupported and RequestedStatesSupported Values (Recommended Equivalence)

PowerStatesSupported Value	RequestedStatesSupported Value
3 (Sleep – Light)	9 (Quiesce)
4 (Sleep – Deep)	9 (Quiesce)
5 (Power Cycle (Off – Soft))	11 (Reset)
6 (Off – Hard)	3 (Disabled)
7 (Hibernate (Off – Soft))	9 (Quiesce)
9 (Power Cycle (Off – Hard))	-
10 (Master Bus Reset)	11 (Reset)
11 (Diagnostic Interrupt (NMI))	-
13 (Off – Hard Graceful)	3 (Disabled)
14 (Master Bus Reset Graceful)	11 (Reset)
15 (Power Cycle (Off – Soft) Graceful)	11 (Reset)
16 (Power Cycle (Off – Hard) Graceful)	-

422 **7.4 Simple Identity Management**

423 A system can represent Account, AccountManagementService, Group, and Identity. If these entities are
 424 represented for the system, the instrumentation should be conformant with the [Simple Identity](#)
 425 [Management Profile](#). If these entities are instrumented in conformance with the [Simple Identity](#)
 426 [Management Profile](#), the Central Instance of the *Base Desktop and Mobile Profile* shall be associated
 427 with the Central Instance of the [Simple Identity Management Profile](#) through the CIM_HostedService
 428 association.

429 **7.5 Role Based Authorization**

430 A system can represent Role, RoleBasedAuthenticationService and Privilege. If these entities are
 431 represented for the system, the instrumentation should be conformant with the [Role Based Authorization](#)
 432 [Profile](#). If these entities are instrumented in conformance with the [Role Based Authorization Profile](#), the
 433 Central Instance of the *Base Desktop and Mobile Profile* shall be associated with the Central Instance of
 434 the [Role Based Authorization Profile](#) through the CIM_HostedService association.

435 **8 Methods**

436 All intrinsic and extrinsic methods are supported as defined in the [Computer System Profile](#).

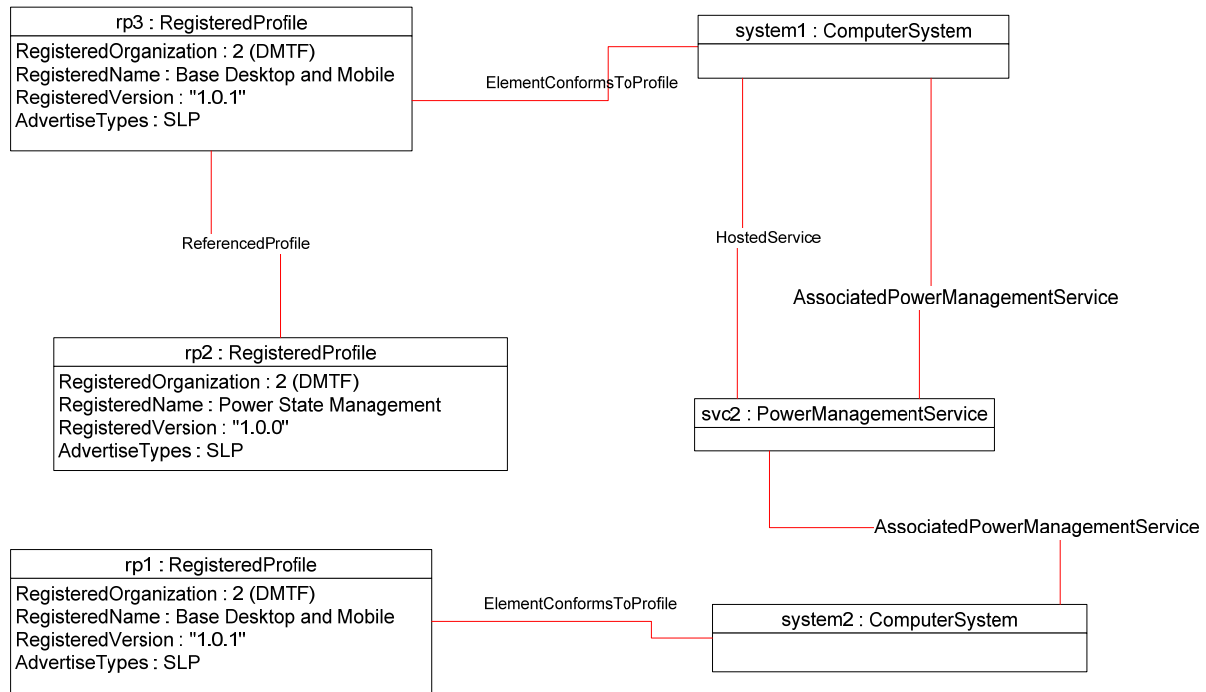
437 **9 Use Cases**

438 The following object diagrams and use cases are based on the implementation conforming to the *Base*
 439 *Desktop and Mobile Profile*.

440 **9.1 Object Diagrams**

441 Figure 2 shows two systems conformant with the *Base Desktop and Mobile Profile*. Both rp3 and rp1
 442 advertise the instrumentation of the *Base Desktop and Mobile Profile*. rp2 advertises the existence of the
 443 [Power State Management Profile](#). rp2 is associated with rp3, which is an instance of
 444 CIM_RegisteredProfile that advertises the *Base Desktop and Mobile Profile*. System1 provides power
 445 control over itself and system2. The ability to provide power control is modeled by svc2. The [Power State](#)

446 [Management Profile](#) is advertised as supported on system1 because that is where the functionality is
 447 accessible.



448

449

Figure 2 – Profile Registration

450 Figure 3 shows the power management functionality available to system1 and system2. Each system
 451 hosts an instance of CIM_TimeService for managing the system’s time. System1 has been configured to
 452 power on at 8 A.M. EST on August 13, 2006, as indicated by the value of the PowerOnTime property of
 453 the instance of CIM_AssociatedPowerManagementService that references system1. This value is relative
 454 to the system time as returned by a call to the ManageTime() method of svc1. Note that state
 455 management is supported with functional equivalence to the supported power state management.
 456 System2 is off and is not configured to come back on.

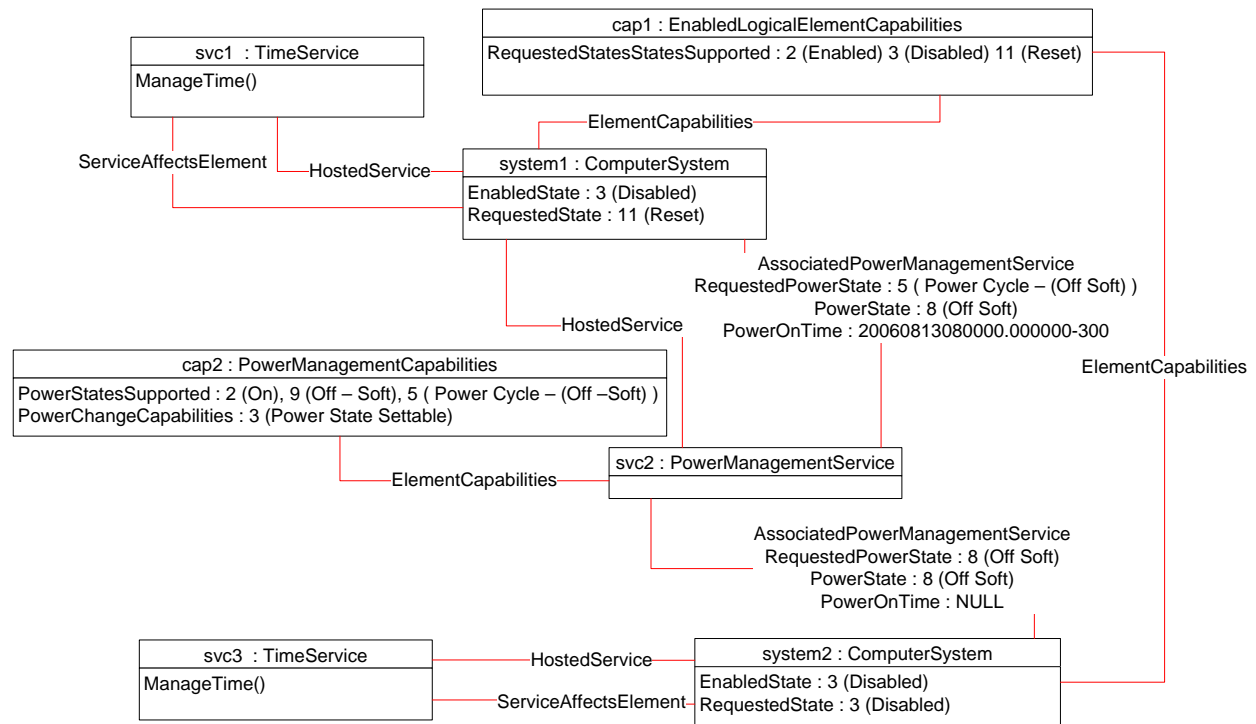


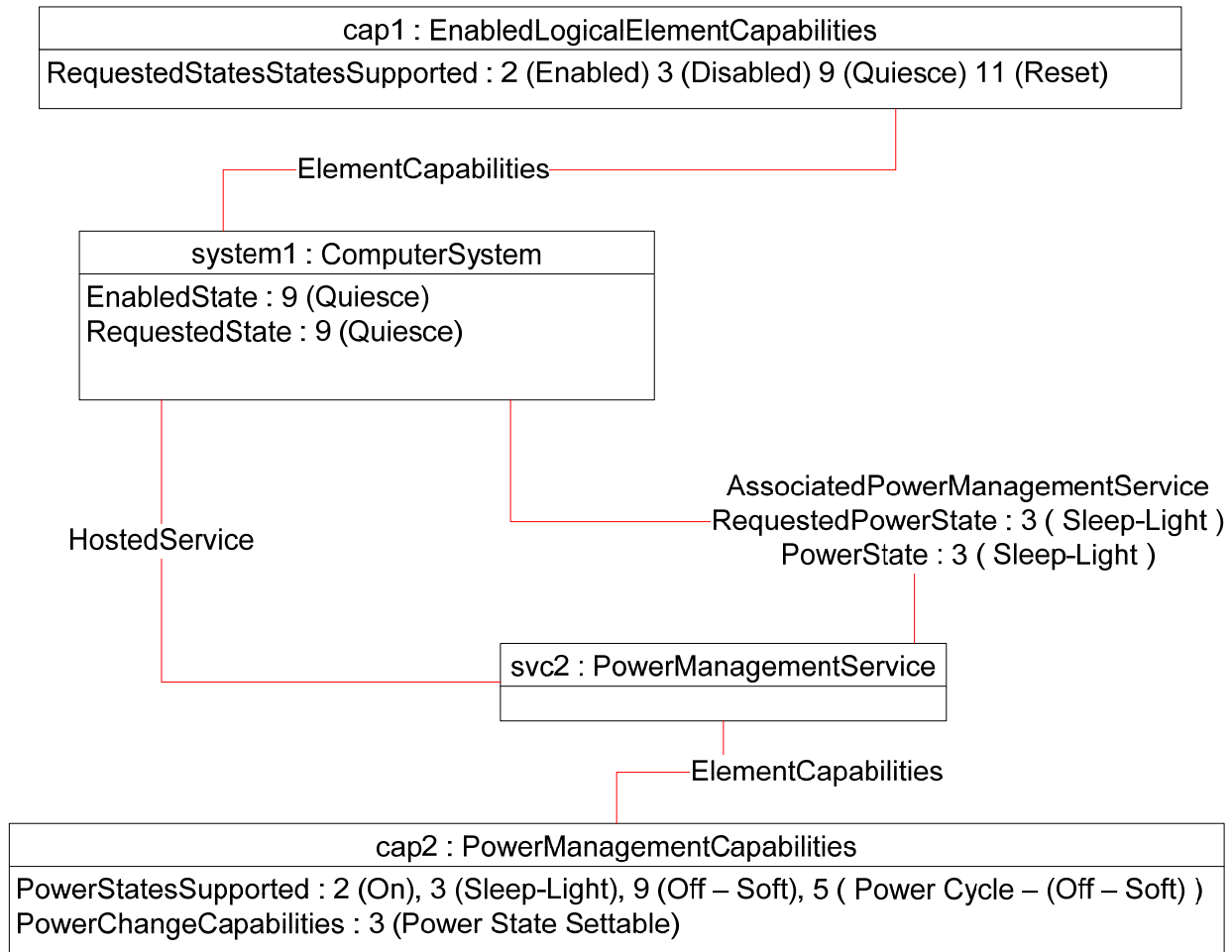
Figure 3 – Power Management and Time Service

457

458

459 Figure 4 shows a system in which the ability to put the system into a sleep-light power state is supported.
 460 The sleep-light state is an extended power state that is expressible through the
 461 CIM_ComputerSystem.EnabledState property. The CIM_ComputerSystem.EnabledState property has
 462 the value 9 (Quiesce) because the current power state of the system is sleep light. If the power state was
 463 not sleep light and the current power state mapped to another valid EnabledState value, the
 464 EnabledState property would be that value. The actual power state of the system is expressed through
 465 the CIM_AssociatedPowerManagementService.PowerState property.

466



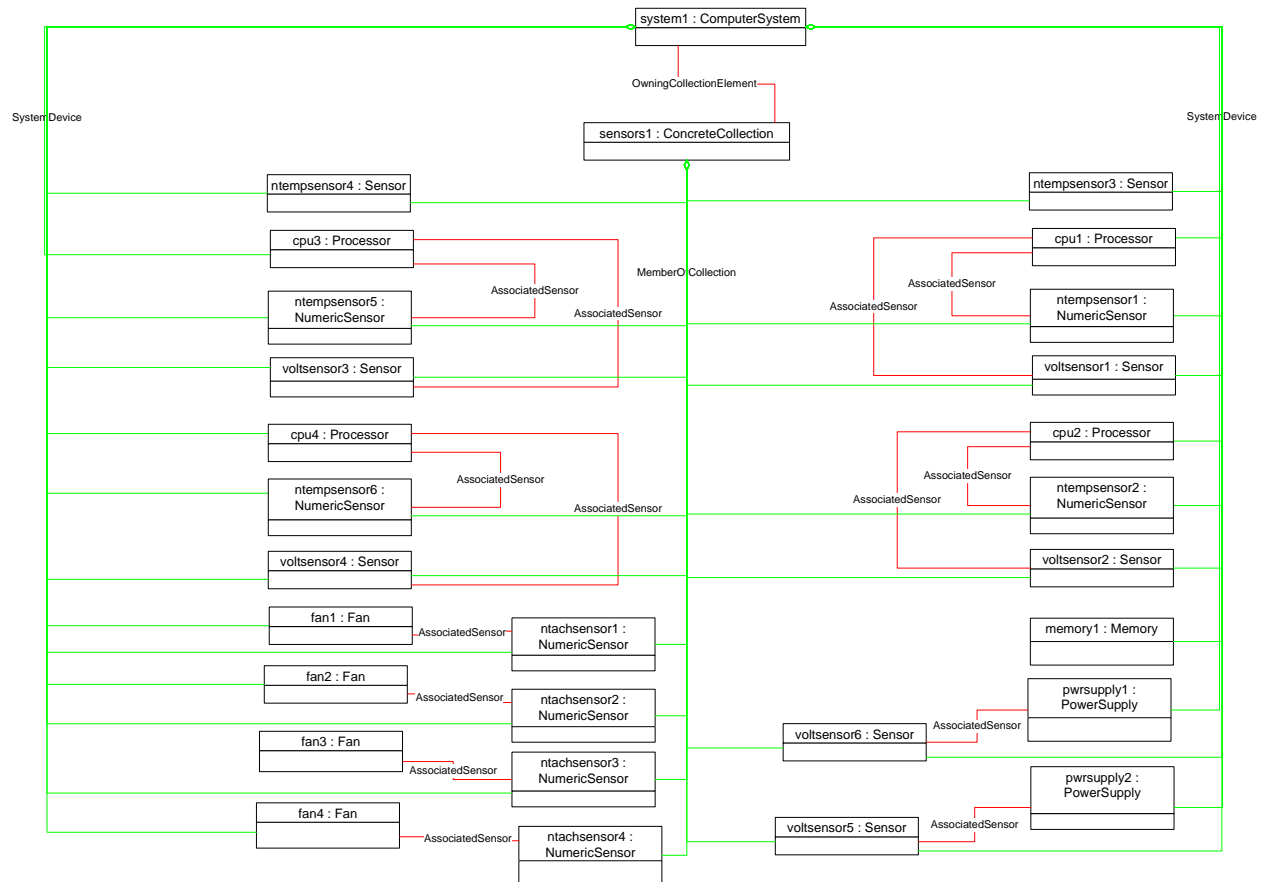
467

468

Figure 4 – Advanced Power Management

469 Figure 5, Figure 6, and Figure 7 illustrate the logical and physical containment hierarchy of a single
 470 system.

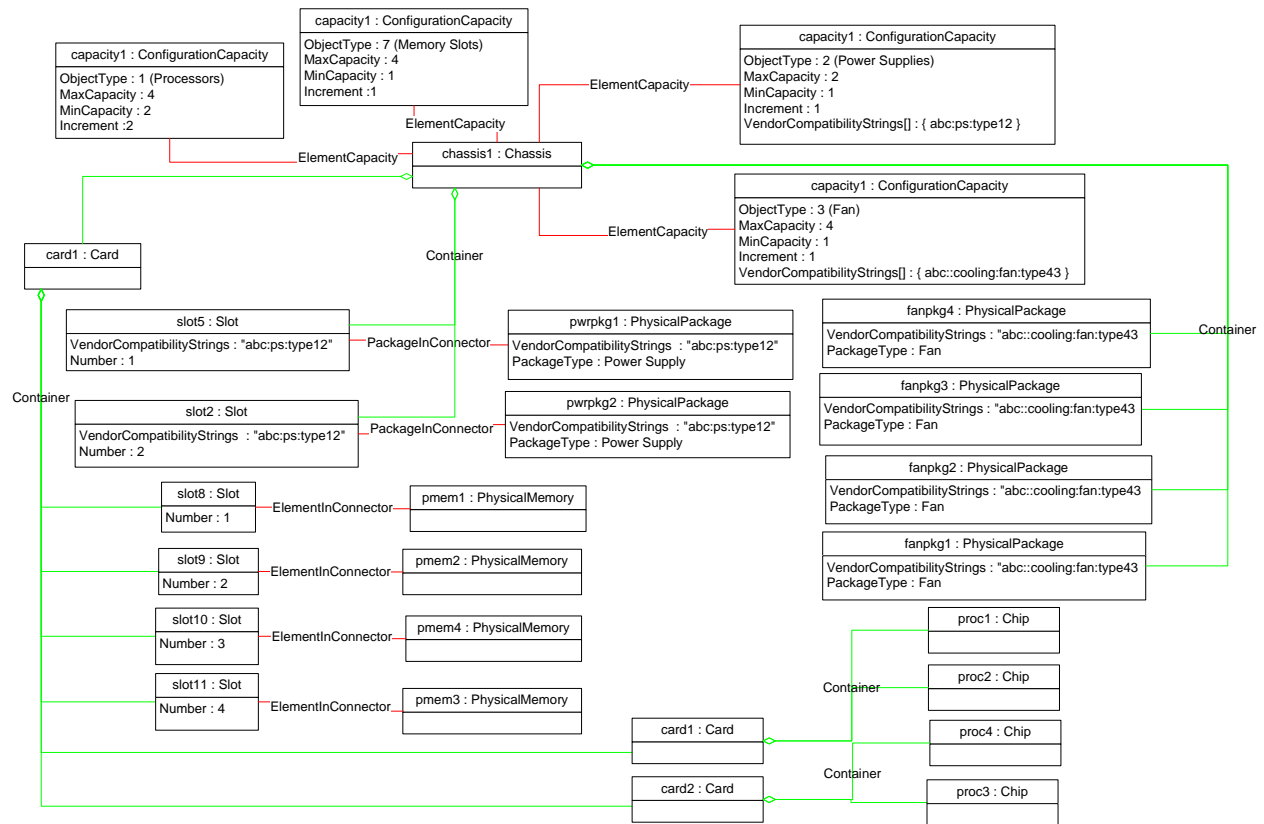
471 Figure 5 illustrates the logical hierarchy of components contained in the system. The optional [CPU](#)
 472 [Profile](#), [Fan Profile](#), [Power Supply Profile](#), [Sensors Profile](#), and [System Memory Profile](#) have been
 473 implemented. The system has four processors. Each processor has a dedicated voltage sensor and a
 474 dedicated temperature sensor. The system has two power supplies. Each power supply has a dedicated
 475 voltage sensor. The system has four fans. Each fan has a dedicated tachometer associated with it. The
 476 total system memory available is modeled as well.



477
478

Figure 5 – Logical Topology

479 Figure 6 shows the physical containment hierarchy for the managed system. The [Physical Asset Profile](#)
 480 has been implemented. The location of the fans within the system is not modeled; instead, they are
 481 modeled as being directly contained in the main system chassis. The slots or bays in the main chassis
 482 that can contain a power supply are separately modeled (slot5 and slot2). The optional slot and package
 483 compatibility behavior of the [Physical Asset Profile](#) has been implemented for the power supply slots. The
 484 system memory is installed in four slots on the main system board (card1). The processors (proc1 –
 485 proc4) are installed in pairs on separate cards on the main system card. The capacity of the system for
 486 processors, fans, power supplies, and memory is indicated through instances of
 487 CIM_ConfigurationCapacity.



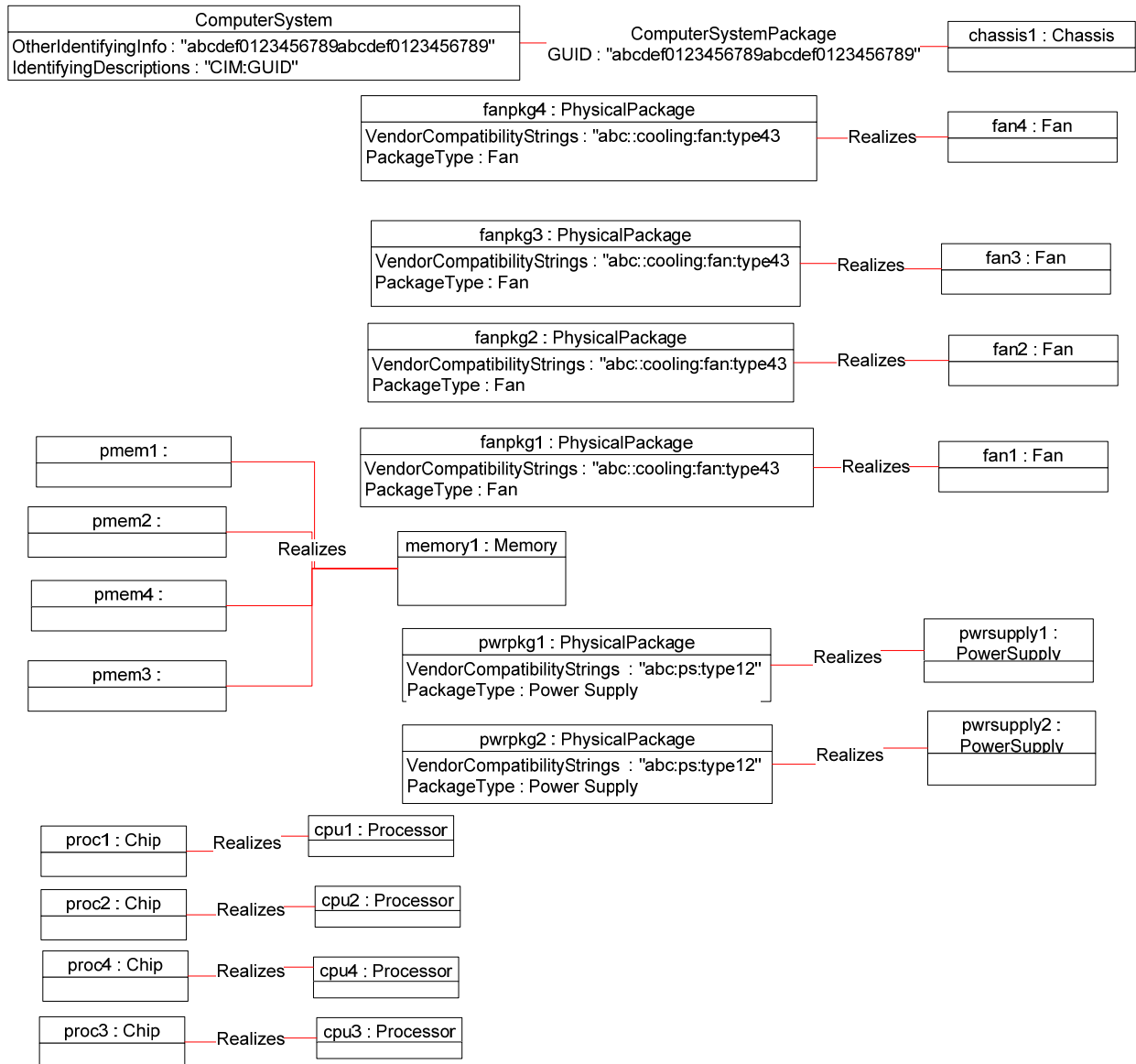
488

489

Figure 6 – Physical Topology

490 Figure 7 shows the relationship between the logical components and their underlying physical packaging.
 491 Each fan, power supply, and processor has a dedicated package. The system memory is realized with
 492 four physical components. The system itself is packaged in a single chassis. To minimize clutter in the
 493 diagram, the CIM_SystemDevice associations have been elided.

494



495

496

Figure 7 – Logical to Physical Mapping

497 **9.2 Determine the System Model and Serial Number**

498 When the *Physical Asset Profile* and optional asset management have been implemented for the system,
 499 a client can determine the system model and serial number as follows:

- 500 1) Find an instance of CIM_PhysicalPackage that is associated with the Central Instance through
 501 the CIM_ComputerSystemPackage association.
 502 2) Refer to the Model and SerialNumber properties of the instance.

503 9.3 Power On a System

504 A client can power on a system as follows:

- 505 1) Look for an instance of CIM_EnabledLogicalElementCapabilities that is associated with the
506 target instance through the CIM_ElementCapabilities association.
- 507 2) Verify that the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property
508 contains the value 2 (Enabled).
- 509 3) Invoke the RequestStateChange() method on the target instance, specifying 2 (Enabled) for the
510 RequestedState parameter.

511 9.4 Power Off a System

512 A client can power off a system as follows:

- 513 1) Look for an instance of CIM_EnabledLogicalElementCapabilities that is associated with the
514 target instance through the CIM_ElementCapabilities association.
- 515 2) Verify that the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property
516 contains the value 3 (Disabled).
- 517 3) Invoke the RequestStateChange() method on the target instance, specifying 3 (Disabled) for
518 the RequestedState parameter.

519 9.5 Shut Down and Restart a System

520 A client can shut down and restart a system as follows:

- 521 1) Look for an instance of CIM_EnabledLogicalElementCapabilities that is associated with the
522 target instance through the CIM_ElementCapabilities association.
- 523 2) Verify that the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property
524 contains the value 11 (Reset).
- 525 3) Invoke the RequestStateChange() method on the target instance, specifying 11 (Reset) for the
526 RequestedState parameter.

527 9.6 Perform System Power Control

528 A client might need to perform power control that is more granular than the functionality available through
529 state management. This power control is done through power state management. A client can determine
530 whether power state management is available for the system by searching for an instance of
531 CIM_PowerManagementService that is associated with the Central Instance through the
532 CIM_AssociatedPowerManagementService association. The specific use cases for performing power
533 state management are documented in the [Power State Management Profile](#).

534 9.7 Determining the System Power State

535 A client can determine the power state of the system as follows:

- 536 1) Query the CIM_ComputerSystem.EnabledState property.
537 If the property has the value 2 (Enabled), the system is currently in ACPI state S0 (or equivalent
538 in a non-ACPI system). If the property has the value 3 (Disabled), the system is currently in
539 ACPI state S5 (or equivalent in a non-ACPI system).
- 540 2) If the CIM_ComputerSystem.EnabledState property has the value 5 (Not Applicable), find the
541 instance of CIM_AssociatedPowerManagementService that references the
542 CIM_ComputerSystem instance.

543 3) Query the value of the CIM_AssociatedPowerManagementService.PowerState property. The
 544 [Power State Management Profile](#) details the equivalent ACPI states for each value.

545 **9.8 Determine the Number of Processors in the System**

546 When the optional [CPU Profile](#) is implemented, the client can determine the number of processors in the
 547 system by querying for instances of CIM_Processor that are associated with the Central Instance through
 548 the CIM_SystemDevice association.

549 The client can use these same steps to find the fans and power supplies installed in the system,
 550 substituting the [Fan Profile](#) and CIM_Fan, and the [Power Supply Profile](#) and CIM_PowerSupply,
 551 appropriately.

552 **9.9 Determine the Number of Processors That the System Can Hold**

553 When the optional configuration capacity behavior from the [Physical Asset Profile](#) is implemented for
 554 processors for the system, a client can determine the number of processors that the system can hold as
 555 follows:

- 556 1) Find instances of CIM_PhysicalPackage that are associated with the Central Instance through
 557 the CIM_ComputerSystemPackage association.
- 558 2) For each instance of CIM_PhysicalPackage, find the instances of CIM_ConfigurationCapacity
 559 that are associated with the CIM_PhysicalPackage instance through the CIM_ElementCapacity
 560 association.
- 561 3) For each instance of CIM_ConfigurationCapacity, if the ObjectType property has the value 1
 562 (Processors), query the MaximumCapacity property and add the value to the total number of
 563 processors that the system can hold.

564 The client can also apply these steps to find the total amount of physical memory, fans, and power
 565 supplies that the system can hold when the configuration capacity has been instrumented for objects of
 566 that type by substituting the appropriate value for 1 (Processors) in step 3.

567 **10 CIM Elements**

568 Table 9 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
 569 implemented as described in Table 9. Clauses 7 (“Implementation Requirements”) and 8 (“Methods”) may
 570 impose additional requirements on these elements.

571 **Table 9 – CIM Elements: Base Desktop and Mobile Profile**

Element Name	Requirement	Description
Classes		
CIM_ComputerSystem	Mandatory	See 10.1.
CIM_ComputerSystemPackage	Mandatory	See 10.2.
CIM_EnabledLogicalElementCapabilities	Optional	See 10.3.
CIM_PhysicalPackage	Mandatory	See 10.4.
CIM_RegisteredProfile	Mandatory	See 10.5.
Indications		
None defined in this profile		

572 10.1 CIM_ComputerSystem

573 An instance of CIM_ComputerSystem is used to represent the system. Table 10 defines the requirements
574 for elements of this class.

575 **Table 10 – Class: CIM_ComputerSystem**

Elements	Requirement	Description
EnabledState	Mandatory	See 7.3.3.1 and 7.3.1.
RequestedState	Mandatory	See 7.3.3.2.
Dedicated	Mandatory	This property shall have the value 32 (“Desktop”) or the value 33 (“Laptop”).

576 10.2 CIM_ComputerSystemPackage

577 One or more instances of CIM_ComputerSystemPackage are used to associate the
578 CIM_ComputerSystem instance with the CIM_PhysicalPackage instances in which it resides. The
579 constraints specified in Table 11 are in addition to those specified in the [Physical Asset Profile](#).

580 **Table 11 – Class: CIM_ComputerSystemPackage**

Elements	Requirement	Description
Dependent	Mandatory	Shall be a reference to the Central Instance Cardinality 1
Antecedent	Mandatory	Shall be a reference to CIM_PhysicalPackage Cardinality 1..*

581 10.3 CIM_EnabledLogicalElementCapabilities

582 CIM_EnabledLogicalElementCapabilities is used to indicate support for managing the state of the system.
583 Table 12 defines the requirements for elements of this class.

584 **Table 12 – Class: CIM_EnabledLogicalElementCapabilities**

Elements	Requirement	Description
RequestedStatesSupported	Mandatory	See 7.3.3.3.

585 10.4 CIM_PhysicalPackage

586 One or more instances of CIM_PhysicalPackage represent the physical packaging of the computer
587 system. Other than the existence of at least one, this profile does not specify any constraints for
588 CIM_PhysicalPackage beyond those specified in the [Physical Asset Profile](#).

589 10.5 CIM_RegisteredProfile

590 CIM_RegisteredProfile identifies the *Base Desktop and Mobile Profile* so that a client can determine
591 whether an instance of CIM_ComputerSystem is conformant with this profile. CIM_RegisteredProfile is
592 defined by the [Profile Registration Profile](#). With the exception of the mandatory values specified for the
593 elements in Table 13, the behavior of the RegisteredProfile instance is per the [Profile Registration Profile](#).

594

Table 13 – Class: CIM_RegisteredProfile

Elements	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of “Base Desktop and Mobile”.
RegisteredVersion	Mandatory	This property shall have a value of “1.0.1”.
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

595
596
597
598

ANNEX A (Informative)

Change Log

Version	Date	Description
1.0.0	2008-12-09	Final release after addressing Platform SC comments
1.0.1	2010-09-15	Final Standard formatted for DMTF Standard release

599
600