



1

2

3

4

Document Number: DSP1008

Date: 2009-06-17

Version: 1.0.0

5 **Modular System Profile**

6 **Document Type: Specification**

7 **Document Status: DMTF Standard**

8 **Document Language: E**

9

10 Copyright Notice

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

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

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26 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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

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

CONTENTS

33	Foreword	5
34	Introduction	6
35	1 Scope	7
36	2 Normative References.....	7
37	2.1 Approved References	7
38	2.2 Other References.....	8
39	3 Terms and Definitions	8
40	4 Symbols and Abbreviated Terms	10
41	5 Synopsis.....	10
42	6 Description	10
43	7 Implementation.....	11
44	7.1 Representing the Modular System	11
45	7.2 Physical Model.....	12
46	7.3 Processor Blades.....	12
47	7.4 Service Processor Profile (Optional).....	13
48	7.5 Power Supply Profile (Optional).....	13
49	7.6 Power Domains (Optional).....	13
50	7.7 Fan Profile (Optional).....	14
51	7.8 Cooling Domains (Optional).....	14
52	7.9 Device Tray Profile (Optional).....	14
53	7.10 Pass-Through Module Profile (Optional)	15
54	7.11 Sensor Profile (Optional)	15
55	8 Methods.....	15
56	8.1 Profile Conventions for Operations.....	15
57	8.2 CIM_AdminDomain.....	15
58	8.3 CIM_ComputerSystem.....	15
59	8.4 CIM_ConcreteDependency	16
60	8.5 CIM_SystemComponent.....	16
61	9 Use Cases.....	16
62	9.1 Object Diagrams	16
63	9.2 Find the CIM_ComputerSystem Instance for the Enclosure	22
64	9.3 Query Chassis Capacity	23
65	9.4 Query Chassis Component Presence	23
66	9.5 Query Chassis Manager Presence.....	23
67	9.6 Find All Power Domains for the Modular System	23
68	9.7 Determine the Power Supply for a Component.....	24
69	9.8 Find All Cooling Domains for the Modular System.....	24
70	9.9 Determine the Fan for a Component	25
71	10 CIM Elements.....	26
72	10.1 CIM_AdminDomain—Power Domain	26
73	10.2 CIM_AdminDomain—Cooling Domain	26
74	10.3 CIM_Chassis.....	27
75	10.4 CIM_ComputerSystem.....	27
76	10.5 CIM_ComputerSystemPackage	27
77	10.6 CIM_ConcreteDependency	28
78	10.7 CIM_PhysicalPackage.....	28
79	10.8 CIM_RegisteredProfile.....	28
80	10.9 CIM_SystemComponent—Cooling Domains	29
81	10.10 CIM_SystemComponent—Power Domains	29
82	10.11 CIM_SystemComponent—Chassis Manager.....	29
83	10.12 CIM_SystemComponent—Processor Blades.....	30
84	ANNEX A (informative) Change Log.....	31

85

86 **Figures**

87	Figure 1 – Modular System Profile: Class Diagram	11
88	Figure 2 – Logical and Physical Topology	17
89	Figure 3 – Chassis Capacity and Compatibility	18
90	Figure 4 – Power Domain	19
91	Figure 5 – Power Management Hosted on Chassis Manager	20
92	Figure 6 – Text Console Redirection Hosted on Chassis Manager	21
93	Figure 7 – Registered Profile	22
94		

95 **Tables**

96	Table 1 – Referenced Profiles	10
97	Table 2 – Operations: CIM_ConcreteDependency	16
98	Table 3 – Operations: CIM_SystemComponent	16
99	Table 4 – Required CIM Elements: Modular System Profile	26
100	Table 5 – Class: CIM_AdminDomain – Power Domain	26
101	Table 6 – Class: CIM_AdminDomain – Cooling Domain	26
102	Table 7 – Class: CIM_Chassis	27
103	Table 8 – Class: CIM_ComputerSystem	27
104	Table 9 – Class: CIM_ComputerSystemPackage	27
105	Table 10 – Class: CIM_ConcreteDependency	28
106	Table 11 – Class: CIM_PhysicalPackage	28
107	Table 12 – Class: CIM_RegisteredProfile	28
108	Table 13 – Class: CIM_SystemComponent – Cooling Domains	29
109	Table 14 – Class: CIM_SystemComponent – Power Domains	29
110	Table 15 – Class: CIM_SystemComponent – Chassis Manager	29
111	Table 16 – Class: CIM_SystemComponent – Processor Blades	30
112		

113

Foreword

114 The *Modular System Profile* (DSP1008) was prepared by the Server Management Working Group and
115 the Physical Platform Profiles Working Group of the DMTF.

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

118

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 • Gary Shippy – IBM
- 131 • Perry Vincent – Intel
- 132 • John Leung – Intel
- 133 • Arvind Kumar – Intel

134

135

Introduction

136 The information in this specification should be sufficient for a provider or consumer of this data to identify
137 unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to
138 represent and manage a blade system that is modeled using the DMTF CIM core and extended model
139 definitions.

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

142

Modular System Profile

143 **1 Scope**

144 The *Modular System Profile* is an autonomous profile for modeling blade systems.

145 **2 Normative References**

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

149 **2.1 Approved References**

150 DMTF DSP0004, *CIM Infrastructure Specification 2.3*,

178 DMTF DSP1024, *Text Console Redirection Profile 1.0*,
179 http://www.dmtf.org/standards/published_documents/DSP1024_1.0.pdf

180 DMTF DSP1025, *Firmware Update Profile 1.0*,
181 http://www.dmtf.org/standards/published_documents/DSP1025_1.0.pdf

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

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

186 **2.2 Other References**

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

189 **3 Terms and Definitions**

190 For the purposes of this document, the terms and definitions in [DSP1033](#) and [DSP1001](#) and the following
191 apply.

192 **3.1**

193 **can**

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

195 **3.2**

196 **cannot**

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

198 **3.3**

199 **conditional**

200 indicates requirements to be followed strictly to conform to the document when the specified conditions
201 are met

202 **3.4**

203 **mandatory**

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

206 **3.5**

207 **may**

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

209 **3.6**

210 **need not**

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

212 **3.7**

213 **optional**

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

215 **3.8**

216 **referencing profile**

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

- 219 **3.9**
220 **shall**
221 indicates requirements to be followed strictly to conform to the document and from which no deviation is
222 permitted
- 223 **3.10**
224 **shall not**
225 indicates requirements strictly to be followed in order to conform to the document and from which no
226 deviation is permitted
- 227 **3.11**
228 **should**
229 indicates that among several possibilities, one is recommended as particularly suitable, without
230 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 231 **3.12**
232 **should not**
233 indicates that a certain possibility or course of action is deprecated but not prohibited
- 234 **3.13**
235 **unspecified**
236 indicates that this profile does not define any constraints for the referenced CIM element or operation
- 237 **3.14**
238 **blade**
239 a physical package that contains one or more operational aspects of a datacenter such as storage,
240 network, or computational functionality, while relying on the containing modular system for infrastructure
241 such as power and cooling
- 242 **3.15**
243 **blade expansion**
244 a physical package that provides additional operational aspects of a computer system to a blade, yet
245 contains insufficient functionality to support an operating system on its own
- 246 **3.16**
247 **cooling domain**
248 the set of systems and components that share a given cooling source that consists of one or more cooling
249 devices
- 250 **3.17**
251 **modular enclosure**
252 the physical packaging of a modular system
- 253 **3.18**
254 **power domain**
255 the set of systems and components that receive power from a given power source that consists of one or
256 more power supplies
- 257 **3.19**
258 **processor blade**
259 a specific type of blade designed to provide processing capability in support of an operating system
- 260 **3.20**
261 **storage blade**
262 a specific type of blade designed to provide storage media or access

263 4 Symbols and Abbreviated Terms

264 None.

265 5 Synopsis

266 **Profile Name:** Modular System

267 **Version:** 1.0.0

268 **Organization:** DMTF

269 **CIM Schema Version:** 2.18

270 **Central Class:** CIM_ComputerSystem

271 **Scoping Class:** CIM_ComputerSystem

272 The *Modular System Profile* extends management capability to include support for blade architectures.
 273 The Central Class of the *Modular System Profile* shall be CIM_ComputerSystem. The Central Instance of
 274 the *Modular System Profile* shall be the instance of CIM_ComputerSystem that represents the modular
 275 system. The Scoping Class for the *Modular System Profile* shall be CIM_ComputerSystem. The Scoping
 276 Instance for the *Modular System Profile* shall be the Central Instance.

277

Table 1 – Referenced Profiles

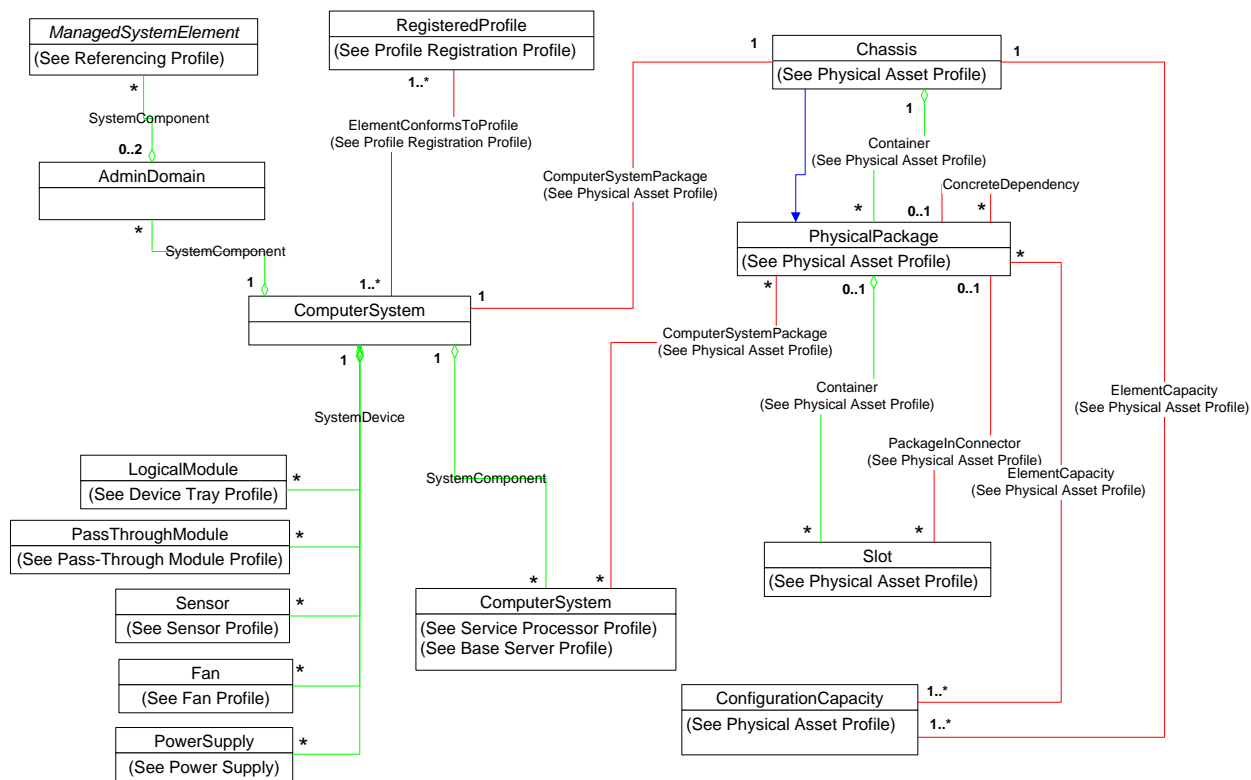
Profile Name	Organization	Version	Relationship	Description
Base Server	DMTF	1.0	Optional	See section 7.3.
Service Processor	DMTF	1.0	Optional	See section 7.4.
Device Tray	DMTF	1.0	Optional	See section 7.9.
Fan	DMTF	1.0	Optional	See section 7.7.
Pass-Through Module	DMTF	1.0	Optional	See section 7.10.
Physical Asset	DMTF	1.0	Mandatory	See section 7.2.
Power Supply	DMTF	1.0	Optional	See section 7.5.
Profile Registration	DMTF	1.0	Mandatory	None.
Sensors	DMTF	1.0	Optional	See section 7.11.

278 6 Description

279 The *Modular System Profile* describes blade systems. Its scope is limited to defining those classes or
 280 behaviors that are unique to blade systems. This profile includes support for the following functionality:

- 281 • representing modular systems, including topology
- 282 • representing the physical packaging of modular systems, including topology
- 283 • modeling power domains of modular systems
- 284 • modeling cooling domains of modular systems

285 Figure 1 represents the class schema for the *Modular System Profile*. For simplicity, the prefix CIM_ has
 286 been removed from the names of the classes.



287

288 **Figure 1 – Modular System Profile: Class Diagram**

289 **7 Implementation**

290 This section details the requirements related to the arrangement of instances and their properties for
 291 implementations of the *Modular System Profile*.

292 The list of all required extrinsic methods and intrinsic operations can be found in section 8 and properties
 293 in section 10.

294 **7.1 Representing the Modular System**

295 The modular system shall be modeled with an instance of CIM_ComputerSystem. It is possible that the
 296 only logical element instrumented will be the Central Instance when no modular components are installed
 297 in the modular system.

298 **7.1.1 Modular Enclosure**

299 A System Chassis, as defined in the *Physical Asset Profile*, shall represent the modular enclosure. An
 300 instance of CIM_ComputerSystemPackage shall reference the CIM_Chassis instance and the Central
 301 Instance.

302 **7.1.2 Scoping a Logical Device**

303 When the implementation uses CIM_LogicalDevice to model a device that is installed into the modular
 304 enclosure and provides function to other components installed in the enclosure, the CIM_LogicalDevice

305 instance shall be associated with the CIM_ComputerSystem instance that represents the modular
306 enclosure through an instance of CIM_SystemDevice.

307 When the instrumentation uses CIM_LogicalDevice to model a component that is part of a chassis
308 manager or processor blade, the CIM_LogicalDevice instance shall be associated with the instance of
309 CIM_ComputerSystem that represents the chassis manager or processor blade through an instance of
310 CIM_SystemDevice.

311 When the instrumentation models a multi-component device that aggregates other devices in the modular
312 enclosure, the multi-component device shall be modeled with an instance of CIM_LogicalModule.

313 **7.2 Physical Model**

314 This section details the requirements for modeling physical aspects of the modular system. The
315 instrumentation shall be conformant with the [Physical Asset Profile](#).

316 One or more instances of CIM_ConfigurationCapacity shall model the capacity of the modular system to
317 contain modular components.

318 An instance of CIM_Slot should exist for each slot or bay of the modular enclosure.

319 **7.3 Processor Blades**

320 The instrumentation of a processor blade shall be conformant with the [Base Server Profile](#). An instance of
321 CIM_SystemComponent shall exist in which the GroupComponent reference is to the Central Instance of
322 this profile and the PartComponent reference is to the Central Instance of the [Base Server Profile](#).

323 **7.3.1 Blade and Blade Expansion Packaging**

324 Implementations shall create at least one instance of CIM_PhysicalPackage for each processor blade
325 installed in the modular chassis. The existence of CIM_PhysicalPackage is conditional on the
326 instrumentation of a CIM_ComputerSystem instance for a processor blade.

327 **7.3.1.1 Blade Physical Package**

328 Implementations shall assign a value of 16 (Blade) to the PackageType property of an instance of
329 CIM_PhysicalPackage when the instance is being used to model a module that can be inserted into a
330 modular chassis and host an operating system.

331 **7.3.1.2 Blade Expansion Physical Package**

332 Implementations shall assign a value of 17 (BladeExpansion) to the PackageType property of an instance
333 of CIM_PhysicalPackage when the instance is being used to model a module that is not stand-alone, is
334 attached to a "Blade" module prior to inserting both modules into the modular chassis, and is an external
335 expansion of the "Blade" module.

336 **7.3.1.3 Relationship between Physical Packages and Slots**

337 When a CIM_PhysicalPackage instance is created to represent a blade module installed in the chassis,
338 the CIM_PhysicalPackage instance should be associated with one instance of CIM_Slot through the
339 CIM_PackageInConnector association. Implementations may associate the CIM_PhysicalPackage
340 instance with more than one instance of CIM_Slot.

341 **7.3.1.4 Relationship between Blade and Blade Expansion**

342 If a CIM_PhysicalPackage instance is created to represent a blade expansion module and the module is
343 connected to a blade module, the implementation shall associate the CIM_PhysicalPackage that
344 represents the blade expansion to the CIM_PhysicalPackage that represents the blade through an

345 instance of CIM_ConcreteDependency. The existence of an instance of CIM_ConcreteDependency is
346 conditional on the existence of an instance of CIM_PhysicalPackage to model a blade expansion.

347 **7.4 Service Processor Profile (Optional)**

348 A modular system may contain one or more chassis managers. When the instrumentation includes
349 support for chassis managers, the chassis managers shall be instrumented compliant with the [Service
350 Processor Profile](#).

351 Each instance of CIM_ComputerSystem that represents a chassis manager shall be associated to the
352 Central Instance through the CIM_SystemComponent association. The GroupComponent property shall
353 be a reference to the Central Instance. The PartComponent property shall be a reference to the
354 CIM_ComputerSystem instance that represents the chassis manager.

355 **7.5 Power Supply Profile (Optional)**

356 When an implementation instruments CIM_PowerSupply to model a power supply in the blade system,
357 the instrumentation shall conform to the [Power Supply Profile](#). When the optional behavior specified in
358 section 7.6 is implemented, for all instances of CIM_SuppliesPower the Dependent reference shall be an
359 instance of CIM_AdminDomain.

360 **7.6 Power Domains (Optional)**

361 A modular system may be responsible for providing power to the modular components installed in it.
362 When a modular system supplies power to modular components, the components may be members of
363 one or more power domains. The power domains of the modular system should be modeled. When the
364 power domains of a modular system are modeled, the requirements detailed in the following subclauses
365 shall be met.

366 **7.6.1 Representing a Power Domain**

367 Exactly one instance of CIM_AdminDomain shall exist for each power domain in the modular system. The
368 instance of CIM_AdminDomain shall be associated with the Central Instance through an instance of
369 CIM_SystemComponent, where the value of the GroupComponent property is the Central Instance and
370 the value of the PartComponent property is the CIM_AdminDomain instance.

371 **7.6.2 Power Supplies in Domain**

372 Each power supply that provides power to the power domain shall be associated with the
373 CIM_AdminDomain instance through an instance of CIM_SuppliesPower. When more than one power
374 supply is able to supply power to the domain, the optional behavior in the "Modeling Power Supply
375 Redundancy" section of the [Base Server Profile](#) should be supported.

376 **7.6.3 Representing Components in a Power Domain**

377 A component is considered to be in a power domain if it receives power from a power supply in the
378 domain. Each instance of a subclass of CIM_LogicalElement that represents a component in a power
379 domain shall be associated with the CIM_AdminDomain instance that represents the domain through the
380 CIM_SystemComponent association. The Central Instance may be associated with the
381 CIM_AdminDomain instance through the CIM_SystemComponent, where the Central Instance is the
382 PartComponent reference. This indicates that components within the modular enclosure that are not
383 explicitly modeled receive power from the domain represented by the CIM_AdminDomain instance.

384 **7.6.4 Representing Slots in a Power Domain**

385 The slots or bays of the modular enclosure that are within a particular power domain may be modeled. A
386 slot or bay is considered to be within a power domain if a component installed in the slot would receive

387 power from the power supply or supplies for the domain. Each instance of CIM_Slot that represents a slot
388 that is in a power domain shall be associated with the CIM_AdminDomain that represents the power
389 domain through the CIM_SystemComponent association.

390 **7.7 Fan Profile (Optional)**

391 If an implementation instruments CIM_Fan to model the cooling functionality of a blade system, the
392 implementation shall conform to the [Fan Profile](#). When the optional behavior specified in section 7.8 is
393 implemented, for each instance of CIM_AssociatedCooling the Dependent reference shall be an instance
394 of CIM_AdminDomain.

395 **7.8 Cooling Domains (Optional)**

396 A modular system may be responsible for providing cooling to the modular components installed in it.
397 When a modular system supplies cooling to modular components, the components may be members of
398 one or more cooling domains. The cooling domains of the modular system should be modeled. When the
399 cooling domains of a modular system are modeled, the requirements detailed in the following subclauses
400 shall be met.

401 **7.8.1 Representing a Cooling Domain**

402 Exactly one instance of CIM_AdminDomain shall exist for each cooling domain in the modular system.
403 The instance of CIM_AdminDomain shall be associated with the Central Instance through the
404 CIM_SystemComponent association, where the value of the GroupComponent property is the Central
405 Instance and the value of the PartComponent property is the CIM_AdminDomain instance.

406 **7.8.2 Fans in Domain**

407 Each instance of CIM_Fan that represents a fan that provides cooling to the cooling domain shall be
408 associated with the CIM_AdminDomain instance through the CIM_AssociatedCooling association. When
409 more than one fan is able to supply cooling to the domain, the optional behavior in the "Modeling Fan
410 Redundancy" section of the [Fan Profile](#) should be supported.

411 **7.8.3 Representing Components in a Cooling Domain**

412 A component is considered to be in a cooling domain if it receives cooling from a fan in the domain. Each
413 instance of a subclass of CIM_LogicalElement that represents a component in a cooling domain shall be
414 associated with the CIM_AdminDomain instance that represents the domain through the
415 CIM_SystemComponent association. The Central Instance may be associated with the
416 CIM_AdminDomain instance through the CIM_SystemComponent association, where the value of the
417 PartComponent property is the Central Instance. This indicates that components within the modular
418 enclosure that are not explicitly modeled receive cooling from the domain represented by the
419 CIM_AdminDomain instance.

420 **7.8.4 Representing Slots in a Cooling Domain**

421 The slots or bays of the modular enclosure that are within a particular cooling domain may be modeled. A
422 slot or bay is considered to be within a cooling domain if a component installed in the slot would receive
423 cooling from the fan or supplies for the domain. Each instance of CIM_Slot that represents a slot that is in
424 a cooling domain shall be associated with the instance of CIM_AdminDomain that represents the cooling
425 domain through the CIM_SystemComponent association.

426 **7.9 Device Tray Profile (Optional)**

427 A modular system may include one or more device trays. When a device tray is modeled, the
428 instrumentation shall be conformant with the [Device Tray Profile](#).

429 7.10 Pass-Through Module Profile (Optional)

430 A modular system may include one or more pass-through modules. When a pass-through module is
431 modeled, the instrumentation shall be in accordance with the requirements specified in the [Pass-Through](#)
432 [Module Profile](#).

433 7.11 Sensor Profile (Optional)

434 If the instrumentation includes support for modeling sensors, the instrumentation shall be conformant with
435 the [Sensors Profile](#).

436 7.11.1 Component Presence Sensors

437 Presence sensors used to determine whether components are installed in slots in the modular system
438 may be modeled using CIM_Sensor. When an instance of CIM_Sensor is used to model a presence
439 sensor for a slot, the CIM_Sensor.SensorType property shall have the value 11 (Presence) and shall be
440 associated with the CIM_Slot instance through the CIM_AssociatedSensor association.

441 8 Methods

442 This section details the requirements for supporting intrinsic operations for the CIM elements defined by
443 this profile. No extrinsic methods exist for the CIM elements specified by this profile.

444 8.1 Profile Conventions for Operations

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

447 The default list of operations is as follows:

- 448 • GetInstance
- 449 • Associators
- 450 • AssociatorNames
- 451 • References
- 452 • ReferenceNames
- 453 • EnumerateInstances
- 454 • EnumerateInstanceNames

455 8.2 CIM_AdminDomain

456 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

458 8.3 CIM_ComputerSystem

459 All operations in the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

461 8.4 CIM_ConcreteDependency

462 Table 2 lists implementation requirements for operations. If implemented, these operations shall be
 463 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 2, all operations in
 464 the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

466 **Table 2 – Operations: CIM_ConcreteDependency**

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

467 8.5 CIM_SystemComponent

468 Table 3 lists implementation requirements for operations. If implemented, these operations shall be
 469 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 3, all operations in
 470 the default list in 8.1 shall be implemented as defined in [DSP0200](#).

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

472 **Table 3 – Operations: CIM_SystemComponent**

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

473 9 Use Cases

474 This section outlines the use cases specific to modular systems. Use cases for functionality that is not
 475 specific to modular systems are documented in the profiles for that functionality. Use cases are
 476 informative and are not intended to define the requirements for conformance.

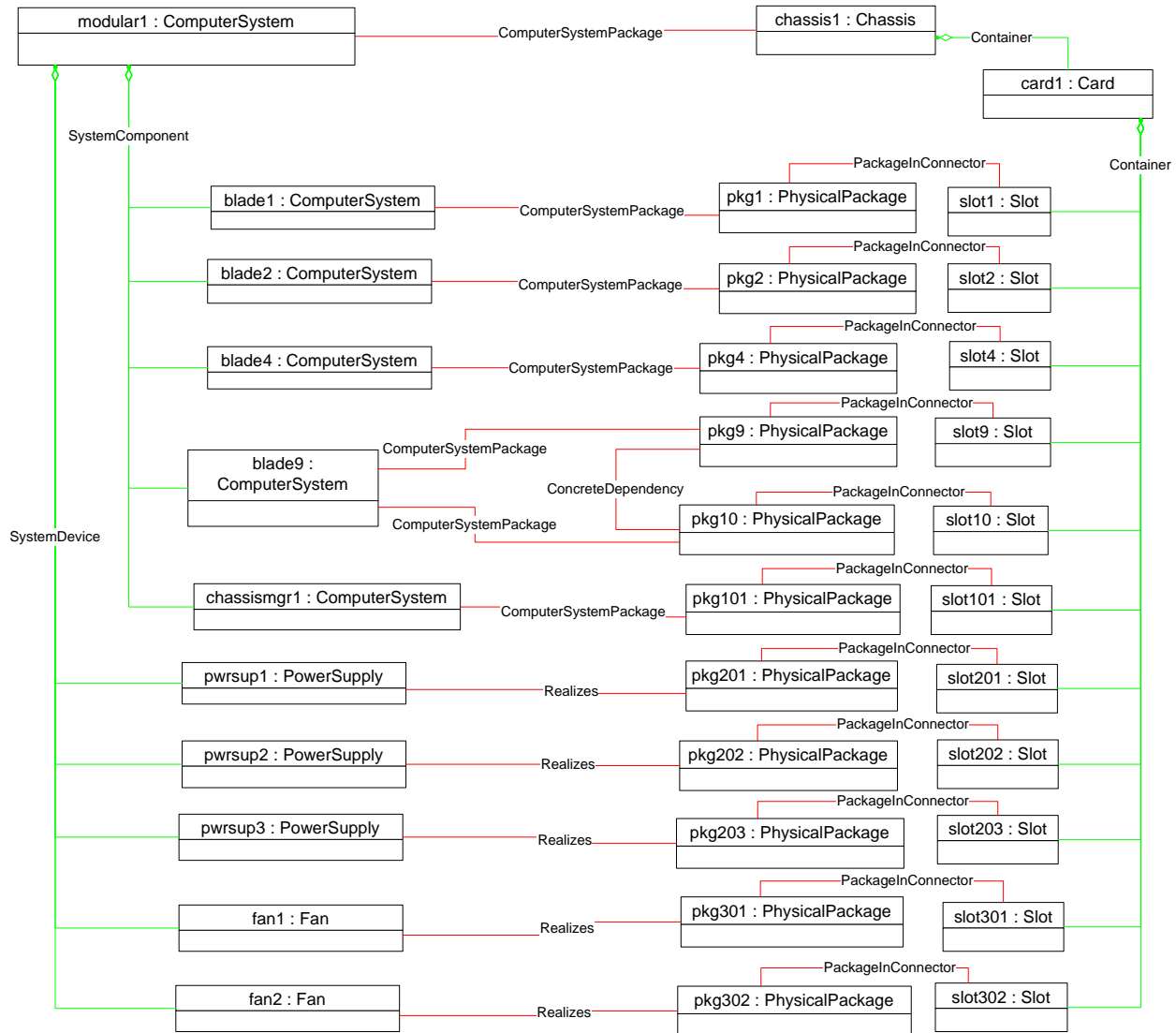
477 9.1 Object Diagrams

478 Figure 2 through Figure 7 are object diagrams that represent a possible instantiation of the *Modular*
 479 *System Profile*.

480 Figure 2 shows the high-level topology of a modular system. The following components are currently
 481 installed in the enclosure:

- 482 • four blade servers
- 483 • three power supplies
- 484 • two fans
- 485 • one chassis manager

486 Blade servers blade1, blade2, and blade4 each consist of a single package installed in a single slot in the
 487 enclosure. blade9 consists of two packages and occupies two slots in the enclosure.

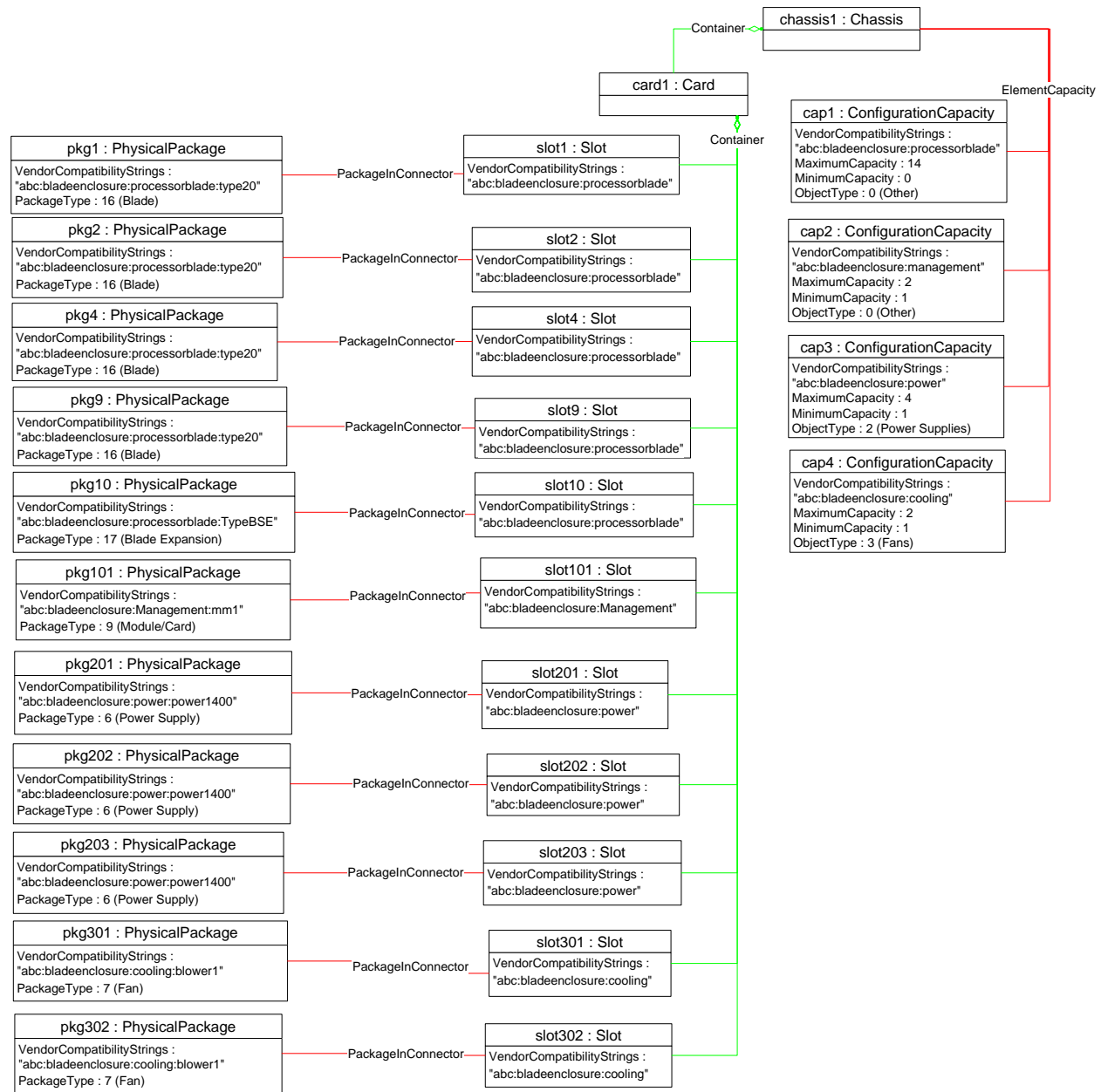


488

489

Figure 2 – Logical and Physical Topology

490 Figure 3 shows the capacity and compatibility of the modular enclosure. Each type of slot is identified with
 491 a unique value for the VendorCompatibilityStrings property of an instance of CIM_Slot, which
 492 corresponds to the value of the VendorCompatibilityStrings property of one of the instances of
 493 CIM_ConfigurationCapacity. For example, the VendorCompatibilityStrings properties of slot1 and cap1
 494 have identical values. Note that an instance of CIM_Slot is not shown in the object diagram for each
 495 possible slot as indicated by the MaximumCapacity property of the instances of
 496 CIM_ConfigurationCapacity. The corresponding instances of CIM_Slot actually exist in the
 497 instrumentation; however, they are not shown to reduce clutter in the diagram. As indicated in Figure 2,
 498 blade9 consists of two packages and occupies two slots in the enclosure. pkg10 is a BladeExpansion
 499 (PackageType = 17) attached to pkg9 (PackageType = 16). An instance of CIM_ConcreteDependency
 500 associates the two instances.



501

502

Figure 3 – Chassis Capacity and Compatibility

503

504

505

506

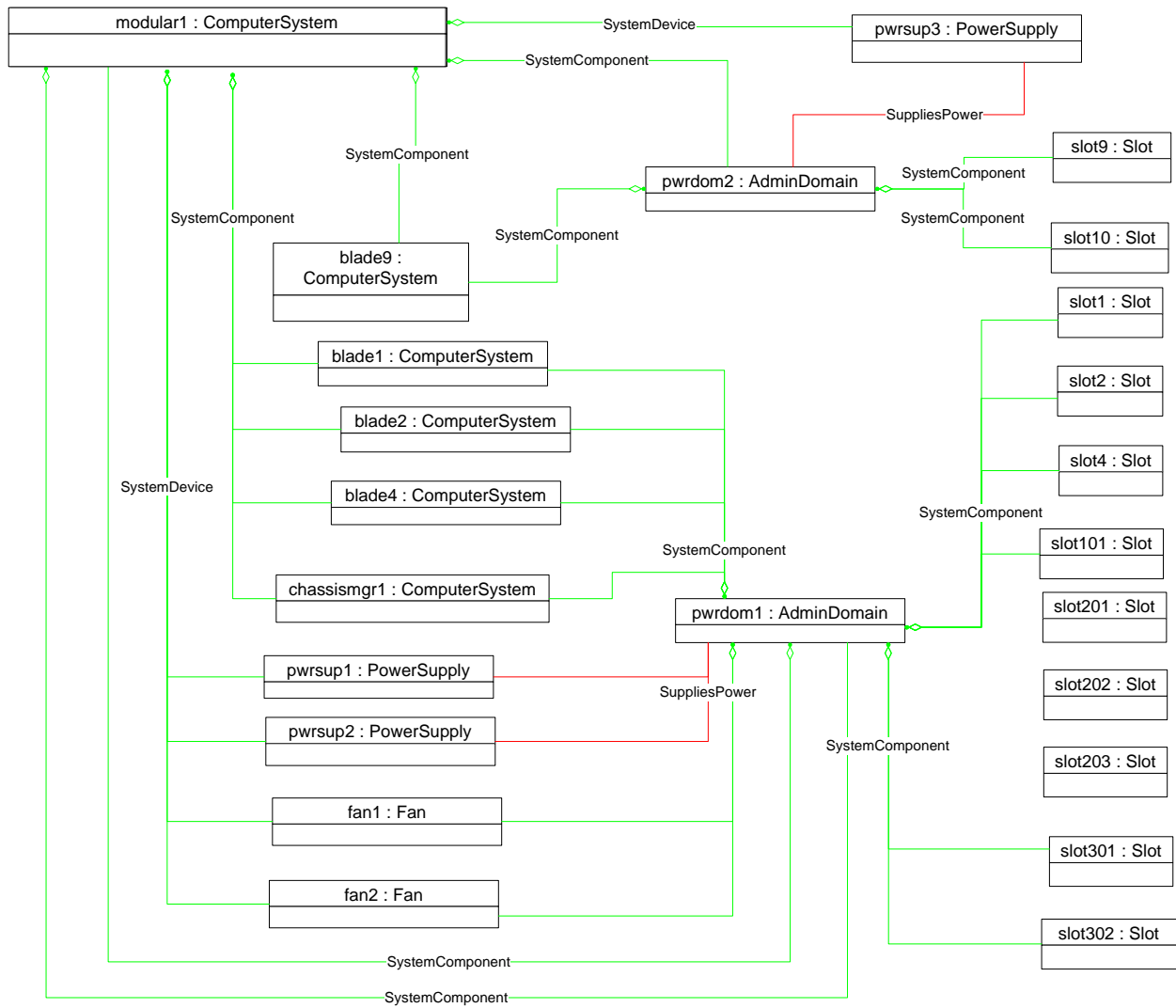
507

508

509

510

Figure 4 illustrates the modeling of power domains in the modular system. Two power domains are modeled, pwrdom1 and pwrdom2. Components in pwrdom1 receive power from power supplies pwr sup1 and pwr sup2, which is indicated by the instances of CIM_SuppliesPower that associate pwr sup1 and pwr sup2 with pwr dom1. Components in pwr dom2 receive power from pwr sup3. Slots that can hold power supplies are not associated with any power domain. The CIM_Slot instances for slots that receive power from the supplies in the domain are always associated with the CIM_AdminDomain instance for the domain through the CIM_SystemComponent association, even when a component has been installed in the slot and is itself associated with the domain.

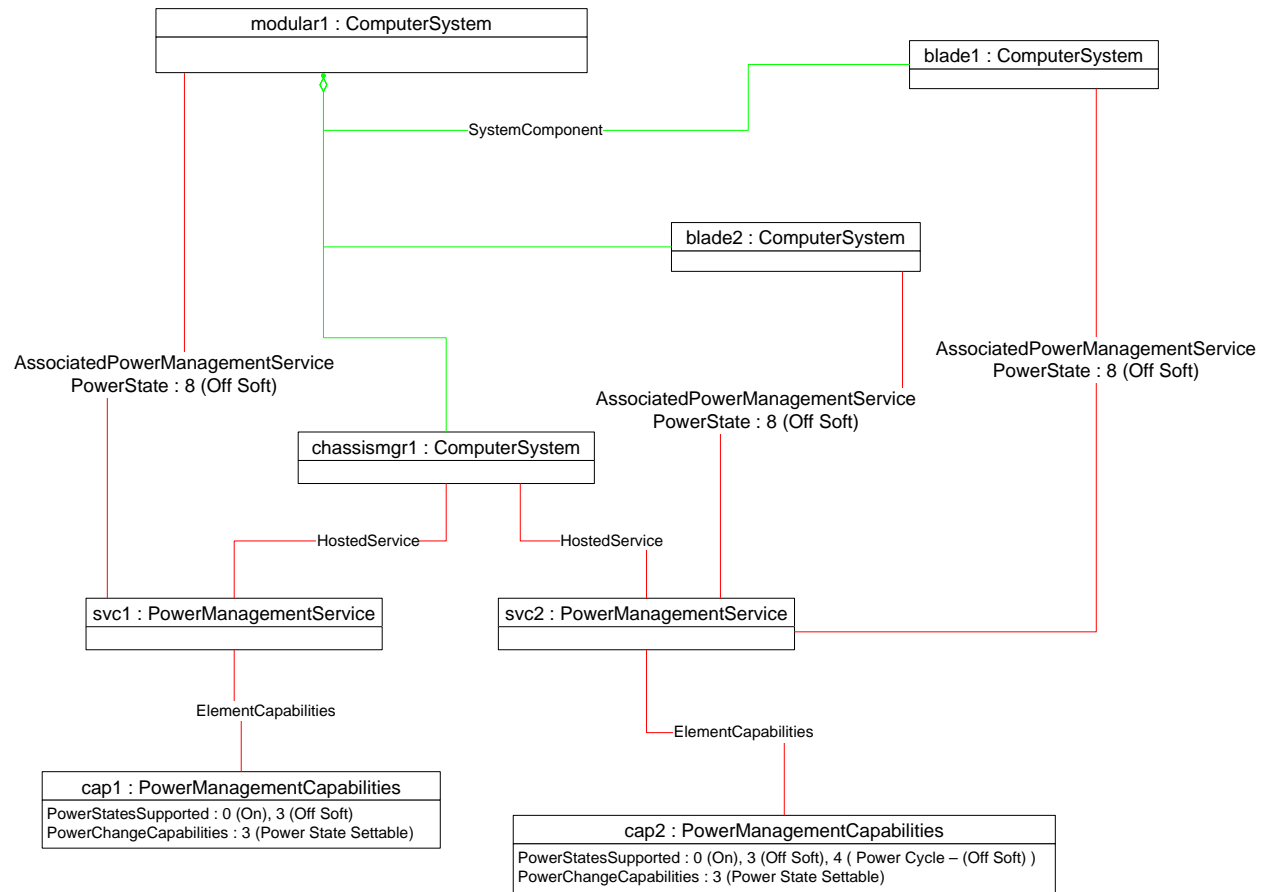


511

512

Figure 4 – Power Domain

513 Figure 5 illustrates power management of the modular system and installed blades that are available
 514 through the installed chassis manager. The modular system and installed blades are all receiving trickle
 515 (flea) power as indicated by the value of the PowerState property for each of the instances of
 516 CIM_AssociatedPowerManagementService. The power management functionality supplied by the
 517 chassis manager is the same for the installed processor blades but distinct for the modular system itself.
 518 Thus, two instances of CIM_PowerManagementService exist, with associated instances of
 519 CIM_PowerManagementCapabilities indicating the functionality available.

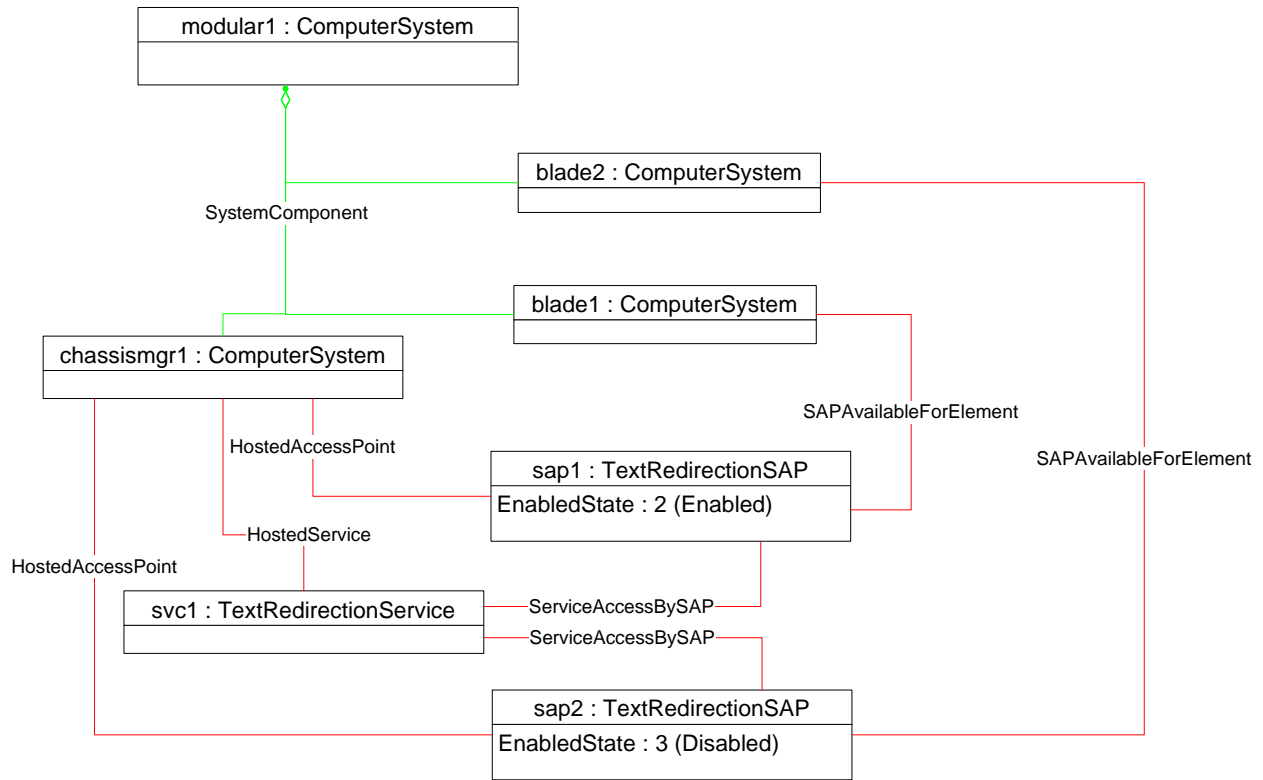


520

521

Figure 5 – Power Management Hosted on Chassis Manager

522 Figure 6 represents the ability of the chassis manager to provide text redirection for the processor blades.
 523 The availability of the function from the chassis manager is indicated by the CIM_HostedService
 524 association between svc1 and chassismgr1. The availability of the function to the processor blades is
 525 indicated by the instances of CIM_SAPAvailableForElement that associate sap1 and sap2 to blade1 and
 526 blade2, respectively.

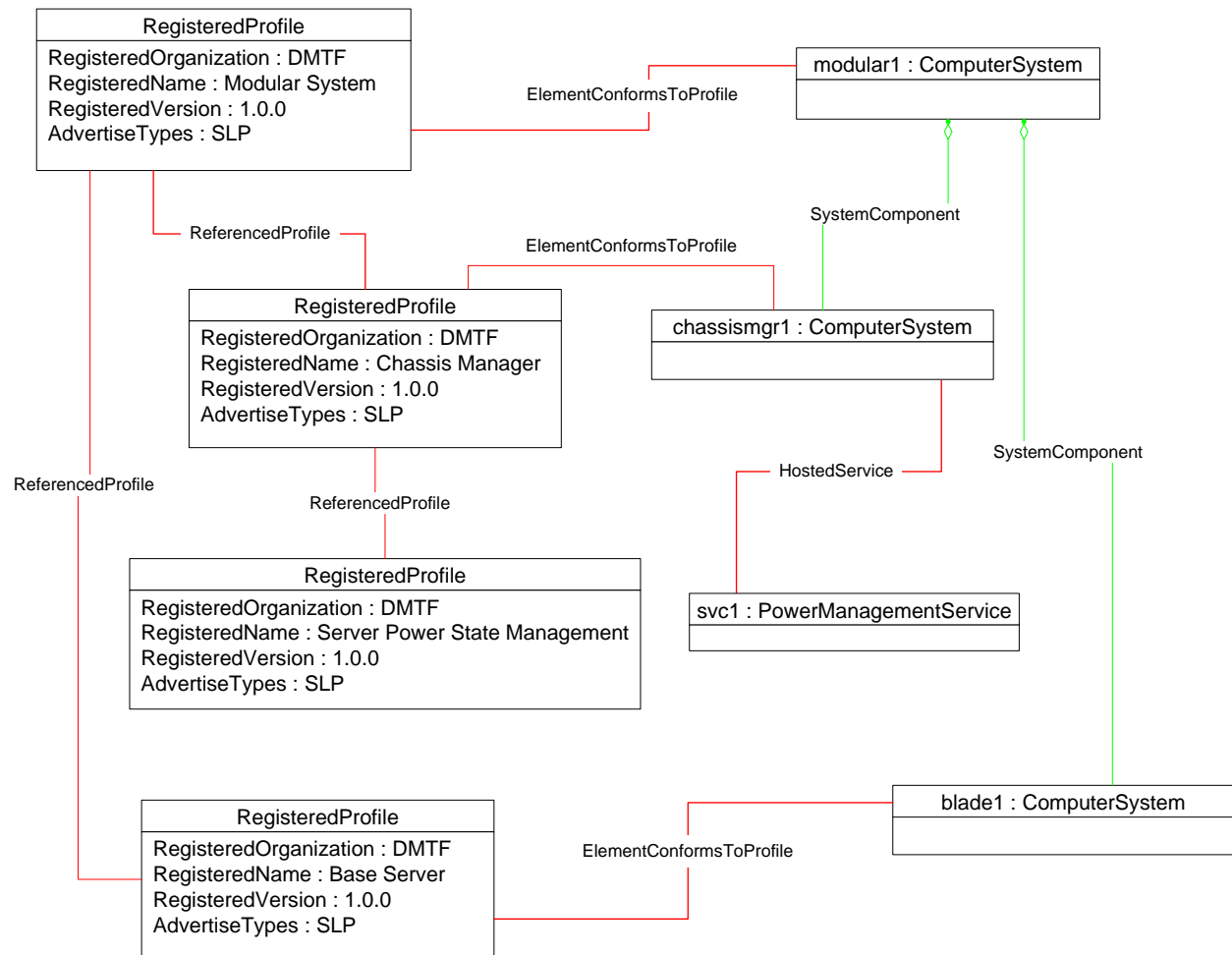


527

528

Figure 6 – Text Console Redirection Hosted on Chassis Manager

529 Figure 7 indicates how an implementation would advertise the implementation of the *Modular System*
 530 *Profile*. The instances of CIM_RegisteredProfile are created in the Interop namespace while the other
 531 instances are created in an Implementation namespace. The *Modular System Profile* and the [Service](#)
 532 [Processor Profile](#) are autonomous profiles. Thus the Central Class Methodology is used. The [Server](#)
 533 [Power State Management Profile](#) is a component profile, and, in this instance, the Scoping Class
 534 Methodology is used.



535

536

Figure 7 – Registered Profile

537 **9.2 Find the CIM_ComputerSystem Instance for the Enclosure**

538 A client can determine whether a modular enclosure is modeled as follows:

- 539 1) Look in the Interop namespace for an instance of CIM_RegisteredProfile that represents this
 540 profile specification.
- 541 2) Look for instances of the CIM_ElementConformsToProfile association that reference the
 542 CIM_RegisteredProfile instance.
- 543 3) Find the CIM_ComputerSystem instance that represents the modular enclosure by traversing
 544 each instance of the CIM_ElementConformsToProfile association to an instance of
 545 CIM_ComputerSystem. These referenced CIM_ComputerSystem instances model modular
 546 enclosures.

547 9.3 Query Chassis Capacity

548 Clients can determine the capacity of the chassis for components of a particular type as follows:

- 549 1) Starting at the CIM_ComputerSystem instance that represents the modular enclosure as found
550 in section 9.2, traverse the CIM_ComputerSystemPackage association to the instance of
551 CIM_Chassis that is the physical side of the model for the modular enclosure.
- 552 2) Use the CIM_ElementCapacity association to find each instance of CIM_ConfigurationCapacity
553 that is associated with the CIM_Chassis instance.
- 554 3) Query the ObjectType and VendorCompatibilityStrings properties of each
555 CIM_ConfigurationCapacity instance to find the instance that represents the component type of
556 interest.
- 557 4) Query the MinimumCapacity and MaximumCapacity properties to determine the capacity of the
558 enclosure for the component type.

559 9.4 Query Chassis Component Presence

560 A client can determine which components are currently installed in the chassis as follows:

- 561 1) Find the CIM_ComputerSystem instance that represents the modular enclosure as specified in
562 section 9.2.
- 563 2) Find all instances of the CIM_SystemComponent association (or subclass) that reference the
564 CIM_ComputerSystem instance, where a reference to the CIM_ComputerSystem instance is
565 the value of the GroupComponent.
- 566 3) Traverse each association instance to the referenced CIM_ManagedSystemElement. The
567 referenced CIM_ManagedSystemElement represents a component installed in the enclosure.
- 568 4) For components that have a corresponding physical presence, if the implementation has
569 instrumented the physical side of the model, find instances of the CIM_Realizes association that
570 reference the CIM_ManagedSystemElement instance.
- 571 5) Traverse the instance of CIM_Realizes to the CIM_PhysicalPackage (or subclass) instance.
- 572 6) If an instance of CIM_PackageInConnector (or subclass) references this instance, determine
573 the slot or connector in which the component is installed.

574 9.5 Query Chassis Manager Presence

575 A client can determine if a Chassis Manager is installed as follows:

- 576 1) Find the CIM_ComputerSystem instance that represents the modular enclosure as described in
577 section 9.2.
- 578 2) Use the steps described in section 9.4 to determine which components are installed in the
579 enclosure, and look for an instance of CIM_ComputerSystem whose Dedicated property
580 contains a value of 29 (Chassis Manager).

581 9.6 Find All Power Domains for the Modular System

582 A client can find all of the power domains for the modular system as follows:

- 583 1) Find instances of CIM_AdminDomain that are associated with the Central Instance through an
584 instance of CIM_SystemComponent whose PartComponent property references the
585 CIM_AdminDomain instance.
- 586 2) For each instance of CIM_AdminDomain, determine if the ElementName property matches
587 "Power Domain".

588 9.7 Determine the Power Supply for a Component

589 When a component is modeled with an instance of a subclass of CIM_ManagedSystemElement, a client
590 can determine the power supply for a component by using the following steps. Note that the algorithm
591 terminates after steps 1, 2, 3, and 4.

- 592 1. Query for an instance of CIM_SuppliesPower that references the CIM_ManagedSystemElement
593 instance.
 - 594 1.1 If one or more such instances exist, the associated instances of CIM_PowerSupply supply
595 power to the CIM_ManagedSystemElement instance.
- 596 2. Query for an instance of CIM_SystemComponent that references the
597 CIM_ManagedSystemElement instance.
 - 598 2.1 If the GroupComponent reference is to an instance of CIM_AdminDomain, query for an
599 instance of CIM_SuppliesPower that references the CIM_AdminDomain instance.
 - 600 2.2 If one or more such instances exist, the associated instances of CIM_PowerSupply supply
601 power to the CIM_ManagedSystemElement instance.
- 602 3. Query for an instance of CIM_SystemComponent (or a subclass) in which the
603 CIM_ManagedSystemElement instance is the value of the PartComponent reference and an
604 instance of CIM_ComputerSystem is the value of the GroupComponent reference.
 - 605 3.1 Find all instances of CIM_PowerSupply that are associated with the CIM_ComputerSystem
606 instance through the CIM_SystemDevice association.
 - 607 3.2 If one or more such instances exist, the associated instances of CIM_PowerSupply supply
608 power to the CIM_ManagedSystemElement instance.
- 609 4. If the instance of CIM_ManagedSystemElement is an instance of CIM_Slot, complete the following
610 steps:
 - 611 4.1 Follow the CIM_Container or CIM_PackageInSlot associations to an instance of
612 CIM_PhysicalElement that represents an outer container.
 - 613 4.1.1 If the instance of CIM_PhysicalElement is an instance of CIM_PhysicalPackage or a
614 subclass, query for an instance of CIM_ComputerSystemPackage that references the
615 CIM_PhysicalPackage instance. If not, repeat step 4.1.
 - 616 4.1.2 If such an instance exists, select the CIM_ComputerSystem instance and proceed to
617 step 4.2. If not, repeat step 4.1.
 - 618 4.2 Find all instances of CIM_PowerSupply that are associated with the CIM_ComputerSystem
619 instance through the CIM_SystemDevice association.
 - 620 4.3 If one or more such instances exist, the associated instances of CIM_PowerSupply supply
621 power to the CIM_ManagedSystemElement instance.

622 9.8 Find All Cooling Domains for the Modular System

623 A client can find all of the cooling domains for the modular system as follows:

- 624 1) Find instances of CIM_AdminDomain that are associated with the Central Instance through an
625 instance of CIM_SystemComponent whose PartComponent property references the
626 CIM_AdminDomain instance.
- 627 2) For each instance of CIM_AdminDomain, determine if the ElementName property matches
628 "Cooling Domain".

629 9.9 Determine the Fan for a Component

630 When a component is modeled with an instance of a subclass of CIM_ManagedSystemElement, a client
631 can determine the fan for a component by using the following steps. Note that the algorithm terminates
632 after steps 1, 2, 3, and 4.

- 633 1. Query for an instance of CIM_AssociatedCooling that references the
634 CIM_ManagedSystemElement instance.
 - 635 1.1 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to
636 the CIM_ManagedSystemElement.
- 637 2. Query for an instance of CIM_SystemComponent that references the
638 CIM_ManagedSystemElement instance.
 - 639 2.1 If the GroupComponent reference is to an instance of CIM_AdminDomain, query for an
640 instance of CIM_AssociatedCooling that references the CIM_AdminDomain instance.
 - 641 2.2 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to
642 the CIM_ManagedSystemElement.
- 643 3. Query for an instance of CIM_SystemComponent (or a subclass) in which the
644 CIM_ManagedSystemElement instance is the value of the PartComponent reference and an
645 instance of CIM_ComputerSystem is the value of the GroupComponent reference.
 - 646 3.1 Find all instances of CIM_Fan that are associated with the CIM_ComputerSystem instance
647 through the CIM_SystemDevice association.
 - 648 3.2 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to
649 the CIM_ManagedSystemElement.
- 650 4. If the instance of CIM_ManagedSystemElement is an instance of CIM_Slot, complete the following
651 steps:
 - 652 4.1 Follow the CIM_Container or CIM_PackageInSlot associations to an instance of
653 CIM_PhysicalElement that represents an outer container.
 - 654 4.1.1 If the instance of CIM_PhysicalElement is an instance of CIM_PhysicalPackage or a
655 subclass, query for an instance of CIM_ComputerSystemPackage that references the
656 CIM_PhysicalPackage instance. If not, repeat step 4.1.
 - 657 4.1.1.1 If such an instance exists, select the CIM_ComputerSystem instance and
658 proceed to step 4.2. If not, repeat step 4.1.
 - 659 4.2 Find all instances of CIM_Fan that are associated with the CIM_ComputerSystem instance
660 through the CIM_SystemDevice association.
 - 661 4.3 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to
662 the CIM_ManagedSystemElement instance.

663 **10 CIM Elements**

664 Table 4 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
 665 implemented as described in Table 4. Sections 7 (“Implementation”) and 8 (“Methods”) may impose
 666 additional requirements on these elements.

667 **Table 4 – Required CIM Elements: Modular System Profile**

Element Name	Requirement	Description
Classes		
CIM_AdminDomain	Optional	See 10.1 and 10.2.
CIM_Chassis	Mandatory	See 10.3.
CIM_ComputerSystem	Mandatory	See 10.4.
CIM_ComputerSystemPackage	Mandatory	See 10.5.
CIM_ConcreteDependency	Conditional	See 7.3.1.4 and 10.6.
CIM_PhysicalPackage	Conditional	See 10.7 and 7.3.1.
CIM_RegisteredProfile	Mandatory	See 10.8.
CIM_SystemComponent	Conditional	See 10.9, 10.10, 10.11, and 10.12.
Indications		
None defined in this profile		

668 **10.1 CIM_AdminDomain—Power Domain**

669 CIM_AdminDomain represents power domains of the modular system. Table 5 contains the requirements
 670 for properties of the instance.

671 **Table 5 – Class: CIM_AdminDomain – Power Domain**

Elements	Requirement	Notes
Name	Mandatory	None
CreationClassName	Mandatory	None
ElementName	Mandatory	Matches "Power Domain"

672 **10.2 CIM_AdminDomain—Cooling Domain**

673 CIM_AdminDomain represents cooling domains of the modular system. Table 6 contains the
 674 requirements for properties of the instance.

675 **Table 6 – Class: CIM_AdminDomain – Cooling Domain**

Elements	Requirement	Notes
Name	Mandatory	None
CreationClassName	Mandatory	None
ElementName	Mandatory	Matches "Cooling Domain"

676 **10.3 CIM_Chassis**

677 CIM_Chassis is defined by the [Physical Asset Profile](#). The requirements denoted in Table 7 are in
 678 addition to those mandated by the [Physical Asset Profile](#).

679 **Table 7 – Class: CIM_Chassis**

Elements	Requirement	Notes
MultipleSystemSupport	Mandatory	This property shall have a value of TRUE.

680 **10.4 CIM_ComputerSystem**

681 An instance of CIM_ComputerSystem represents the modular enclosure. Table 8 contains the
 682 requirements for properties of the instance.

683 **Table 8 – Class: CIM_ComputerSystem**

Elements	Requirement	Notes
Dedicated	Mandatory	Matches 0 (Other)
OtherDedicatedDescriptions	Mandatory	Matches "Modular"
Name	Mandatory	None
CreationClassName	Mandatory	None
ElementName	Mandatory	Pattern (".*")
OperationalStatus	Mandatory	None
HealthState	Mandatory	None

684 **10.5 CIM_ComputerSystemPackage**

685 CIM_ComputerSystemPackage associates the CIM_Chassis instance for the modular enclosure with the
 686 CIM_ComputerSystem instance for the modular enclosure. Requirements specified in Table 9 are in
 687 addition to those specified in the [Physical Asset Profile](#).

688 **Table 9 – Class: CIM_ComputerSystemPackage**

Elements	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to an instance of CIM_Chassis that represents the modular enclosure. Cardinality 1
Dependent	Mandatory	This property shall be a reference to the Central Instance. Cardinality 1

689 10.6 CIM_ConcreteDependency

690 CIM_ConcreteDependency associates a blade expansion physical package with a blade physical
691 package. Table 10 contains the requirements for properties of the instance.

692 **Table 10 – Class: CIM_ConcreteDependency**

Elements	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to an instance of CIM_PhysicalPackage that represents the blade. Cardinality 1
Dependent	Mandatory	This property shall be a reference to an instance of CIM_PhysicalPackage that represents the blade expansion. Cardinality *

693 10.7 CIM_PhysicalPackage

694 CIM_PhysicalPackage is defined by the [Physical Asset Profile](#). The requirements denoted in Table 11 are
695 in addition to those mandated by the [Physical Asset Profile](#).

696 **Table 11 – Class: CIM_PhysicalPackage**

Elements	Requirement	Notes
PackageType	Mandatory	See 7.3.1.

697 10.8 CIM_RegisteredProfile

698 CIM_RegisteredProfile identifies the *Modular System Profile* in order for a client to determine whether an
699 instance of CIM_ComputerSystem is conformant with this profile. CIM_RegisteredProfile is defined by the
700 [Profile Registration Profile](#). With the exception of the mandatory values specified for the properties in
701 Table 12, the behavior of the CIM_RegisteredProfile instance is in accordance with the [Profile](#)
702 [Registration Profile](#).

703 **Table 12 – Class: CIM_RegisteredProfile**

Elements	Requirement	Notes
RegisteredName	Mandatory	This property shall have a value of Modular System.
RegisteredVersion	Mandatory	This property shall have a value of "1.0.0".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

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

707 **10.9 CIM_SystemComponent—Cooling Domains**

708 CIM_SystemComponent associates an instance of a sub-class of CIM_ManagedElement with an
 709 instance of CIM_AdminDomain representing the cooling domain in which the element is installed. If no
 710 cooling domains are modeled, or no elements that receive cooling are modeled, no instances of
 711 CIM_SystemComponent exist. Table 13 contains the requirements for properties of the instance. The
 712 existence of CIM_SystemComponent in this context is conditional on the modeling of an element installed
 713 in the cooling domain.

714 **Table 13 – Class: CIM_SystemComponent – Cooling Domains**

Elements	Requirement	Notes
GroupComponent	Mandatory	See 7.8.
PartComponent	Mandatory	See 7.8.

715 **10.10 CIM_SystemComponent—Power Domains**

716 CIM_SystemComponent associates an instance of a sub-class of CIM_ManagedElement with an
 717 instance of CIM_AdminDomain representing the power domain in which the element is installed. If no
 718 power domains are modeled, or no elements that receive power are modeled, no instances of
 719 CIM_SystemComponent exist. Table 14 contains the requirements for properties of the instance. The
 720 existence of CIM_SystemComponent in this context is conditional on the modeling of an element installed
 721 in the power domain.

722 **Table 14 – Class: CIM_SystemComponent – Power Domains**

Elements	Requirement	Notes
GroupComponent	Mandatory	See 7.6.
PartComponent	Mandatory	See 7.6.

723 **10.11 CIM_SystemComponent—Chassis Manager**

724 CIM_SystemComponent associates the CIM_ComputerSystem instance that represents a chassis
 725 manager with the CIM_ComputerSystem instance that represents the modular enclosure in which the
 726 chassis manager is installed. If no chassis managers are modeled, no instances of
 727 CIM_SystemComponent exist. Table 15 contains the requirements for properties of the instance. The
 728 existence of CIM_SystemComponent in this context is conditional on the modeling of a chassis manager.

729 **Table 15 – Class: CIM_SystemComponent – Chassis Manager**

Elements	Requirement	Notes
GroupComponent	Mandatory	See 7.4.
PartComponent	Mandatory	See 7.4.

730 10.12 CIM_SystemComponent—Processor Blades

731 CIM_SystemComponent associates the CIM_ComputerSystem instance that represents a processor
732 blade with the CIM_ComputerSystem instance that represents the modular enclosure in which the
733 processor blade is installed. If no processor blades are modeled, no instances of CIM_SystemComponent
734 exist. Table 16 contains the requirements for properties of the instance. The existence of
735 CIM_SystemComponent in this context is conditional on the modeling of a processor blade.

736 Table 16 – Class: CIM_SystemComponent – Processor Blades

Elements	Requirement	Notes
GroupComponent	Mandatory	See 7.3.
PartComponent	Mandatory	See 7.3.

737
738

739
740
741
742
743

ANNEX A (informative)

Change Log

Version	Date	Description
1.0.0	06-17-2009	DMTF Standard Release

744
745