Document Number: DSP1008

Date: 2009-06-17

Version: 1.0.0

Modular System Profile

Document Type: Specification

Document Status: DMTF Standard

Document Language: E

CONTENTS

85

86 # Figures

94

95 # Tables

112

Foreword

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

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

Acknowledgments

The authors wish to acknowledge the following people.

Editor:

- Aaron Merkin – IBM

Contributors:

- Jon Hass – Dell
- Khachatur Papanyan – Dell
- Enoch Suen – Dell
- Jeff Hilland – HP
- Christina Shaw – HP
- Aaron Merkin – IBM
- Gary Shippy – IBM
- Perry Vincent – Intel
- John Leung – Intel
- Arvind Kumar – Intel

Introduction

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

The target audience for this specification is implementers who are writing CIM-based providers or 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*,
151 http://www.dmtf.org/standards/published_documents/DSP0004_2.3.pdf

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

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

156 DMTF DSP1004, *Base Server Profile 1.0,*
157 http://www.dmtf.org/standards/published_documents/DSP1004_1.0.pdf

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

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

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

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

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

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

170 DMTF DSP1019, *Device Tray Profile 1.0,*
171 http://www.dmtf.org/standards/published_documents/DSP1019_1.0.pdf

172 DMTF DSP1020, *Pass-Through Module Profile 1.0,*
173 http://www.dmtf.org/standards/published_documents/DSP1020_1.0.pdf

174 DMTF DSP1021, *Shared Device Management Profile 1.0,*
175 http://www.dmtf.org/standards/published_documents/DSP1021_1.0.pdf

176 DMTF DSP1023, *Firmware Inventory Profile 1.0,*
177 http://www.dmtf.org/standards/published_documents/DSP1023_1.0.pdf

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

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

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

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

2.2 Other References

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

3 Terms and Definitions

For the purposes of this document, the terms and definitions in DSP1033 and DSP1001 and the following apply.

3.1
can
used for statements of possibility and capability, whether material, physical, or causal

3.2
cannot
used for statements of possibility and capability, whether material, physical, or causal

3.3
conditional
indicates requirements to be followed strictly to conform to the document when the specified conditions are met

3.4
mandatory
indicates requirements to be followed strictly to conform to the document and from which no deviation is permitted

3.5
may
indicates a course of action permissible within the limits of the document

3.6
need not
indicates a course of action permissible within the limits of the document

3.7
optional
indicates a course of action permissible within the limits of the document

3.8
referencing profile
indicates a profile that owns the definition of this class and can include a reference to this profile in its "Referenced Profiles" table

3.9
shall
indicates requirements to be followed strictly to conform to the document and from which no deviation is permitted

3.10
shall not
indicates requirements strictly to be followed in order to conform to the document and from which no deviation is permitted

3.11
should
indicates that among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required

3.12
should not
indicates that a certain possibility or course of action is deprecated but not prohibited

3.13
unspecified
indicates that this profile does not define any constraints for the referenced CIM element or operation

3.14
blade
a physical package that contains one or more operational aspects of a datacenter such as storage, network, or computational functionality, while relying on the containing modular system for infrastructure such as power and cooling

3.15
blade expansion
a physical package that provides additional operational aspects of a computer system to a blade, yet contains insufficient functionality to support an operating system on its own

3.16
cooling domain
the set of systems and components that share a given cooling source that consists of one or more cooling devices

3.17
modular enclosure
the physical packaging of a modular system

3.18
power domain
the set of systems and components that receive power from a given power source that consists of one or more power supplies

3.19
processor blade
a specific type of blade designed to provide processing capability in support of an operating system

3.20
storage blade
a specific type of blade designed to provide storage media or access

4 Symbols and Abbreviated Terms

None.

5 Synopsis

Profile Name: Modular System

Version: 1.0.0

Organization: DMTF

CIM Schema Version: 2.18

Central Class: CIM_ComputerSystem

Scoping Class: CIM_ComputerSystem

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

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.

6 Description

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

- representing modular systems, including topology
- representing the physical packaging of modular systems, including topology
- modeling power domains of modular systems
- modeling cooling domains of modular systems

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

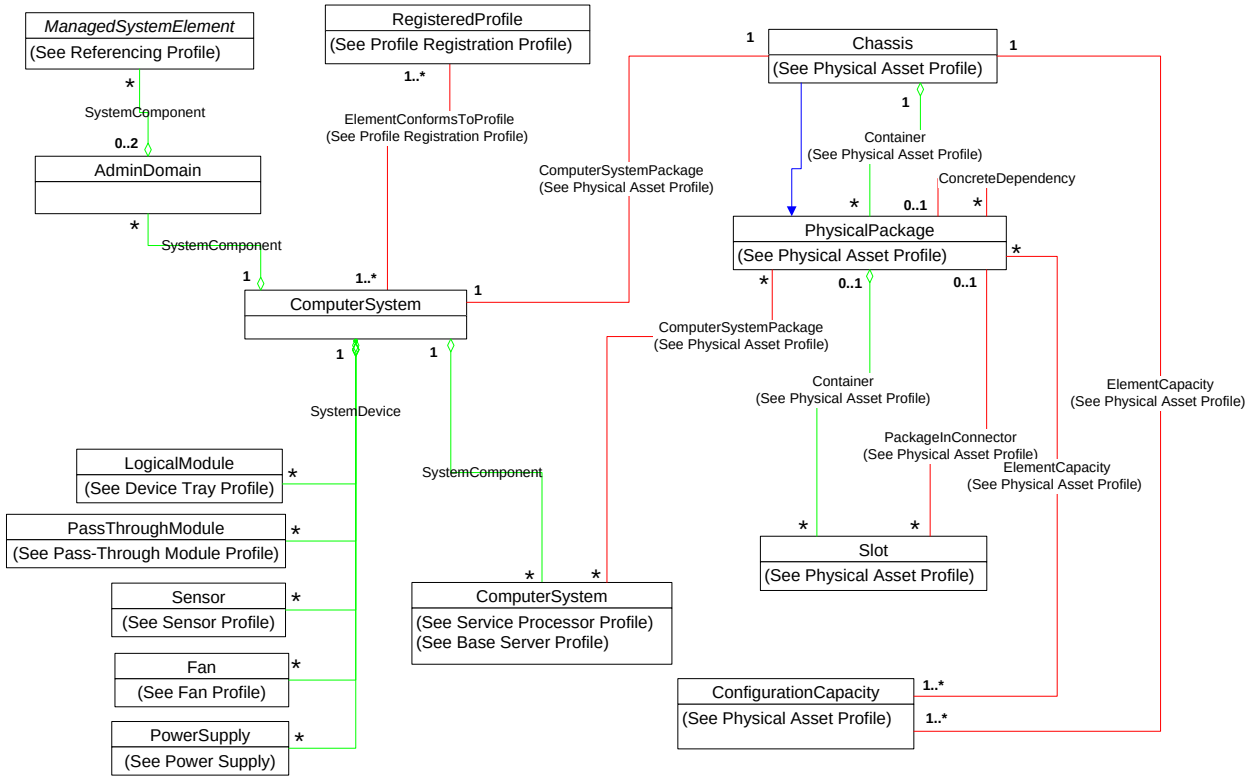


Figure 1 – Modular System Profile: Class Diagram

7 Implementation

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

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

7.1 Representing the Modular System

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

7.1.1 Modular Enclosure

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

7.1.2 Scoping a Logical Device

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

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

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

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

7.2 Physical Model

This section details the requirements for modeling physical aspects of the modular system. The instrumentation shall be conformant with the *Physical Asset Profile*.

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

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

7.3 Processor Blades

The instrumentation of a processor blade shall be conformant with the *Base Server Profile*. An instance of CIM_SystemComponent shall exist in which the GroupComponent reference is to the Central Instance of this profile and the PartComponent reference is to the Central Instance of the *Base Server Profile*.

7.3.1 Blade and Blade Expansion Packaging

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

7.3.1.1 Blade Physical Package

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

7.3.1.2 Blade Expansion Physical Package

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

7.3.1.3 Relationship between Physical Packages and Slots

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

7.3.1.4 Relationship between Blade and Blade Expansion

If a CIM_PhysicalPackage instance is created to represent a blade expansion module and the module is connected to a blade module, the implementation shall associate the CIM_PhysicalPackage that 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

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

7.7 Fan Profile (Optional)

If an implementation instruments CIM_Fan to model the cooling functionality of a blade system, the implementation shall conform to the *Fan Profile*. When the optional behavior specified in section 7.8 is implemented, for each instance of CIM_AssociatedCooling the Dependent reference shall be an instance of CIM_AdminDomain.

7.8 Cooling Domains (Optional)

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

7.8.1 Representing a Cooling Domain

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

7.8.2 Fans in Domain

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

7.8.3 Representing Components in a Cooling Domain

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

7.8.4 Representing Slots in a Cooling Domain

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

7.9 Device Tray Profile (Optional)

A modular system may include one or more device trays. When a device tray is modeled, the instrumentation shall be conformant with the *Device Tray Profile*.

7.10 Pass-Through Module Profile (Optional)

A modular system may include one or more pass-through modules. When a pass-through module is modeled, the instrumentation shall be in accordance with the requirements specified in the *Pass-Through Module Profile*.

7.11 Sensor Profile (Optional)

If the instrumentation includes support for modeling sensors, the instrumentation shall be conformant with the *Sensors Profile*.

7.11.1 Component Presence Sensors

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

8 Methods

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

8.1 Profile Conventions for Operations

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

The default list of operations is as follows:

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

8.2 CIM_AdminDomain

All operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

8.3 CIM_ComputerSystem

All operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

8.4 CIM_ConcreteDependency

Table 2 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 2, all operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

Table 2 – Operations: CIM_ConcreteDependency

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

8.5 CIM_SystemComponent

Table 3 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 3, all operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

Table 3 – Operations: CIM_SystemComponent

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

9 Use Cases

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

9.1 Object Diagrams

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

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

- four blade servers
- three power supplies
- two fans
- one chassis manager

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

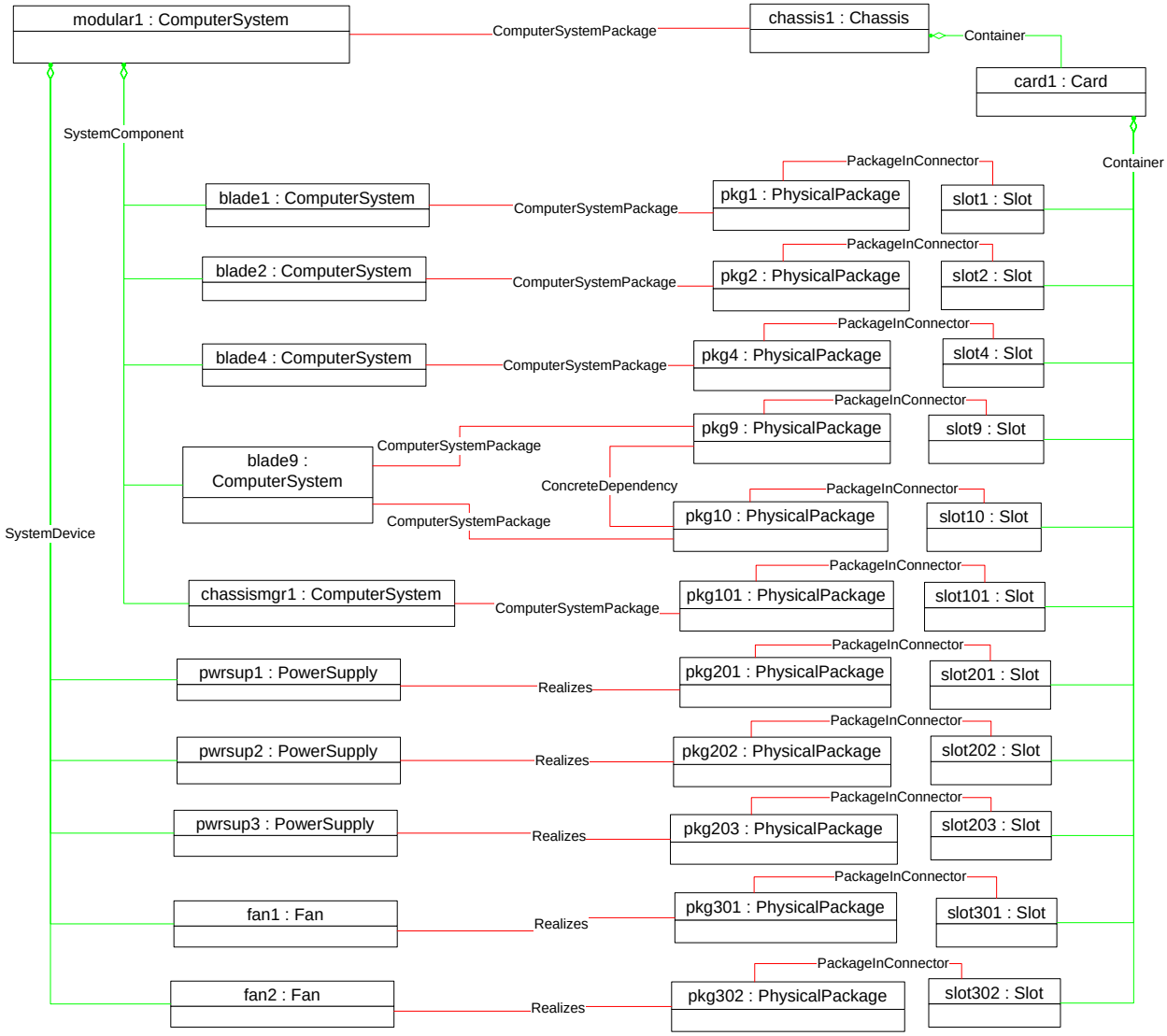


Figure 2 – Logical and Physical Topology

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

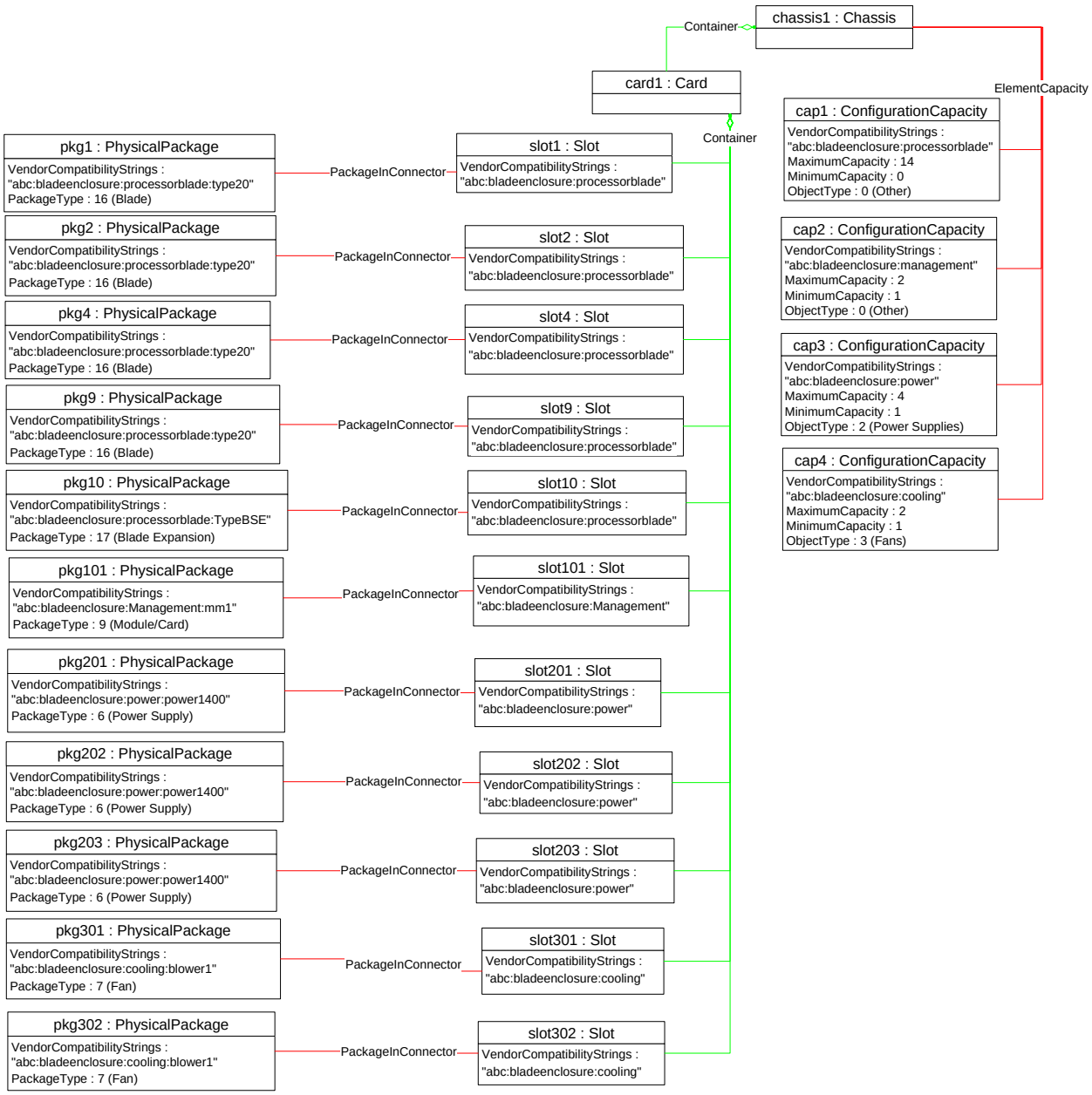
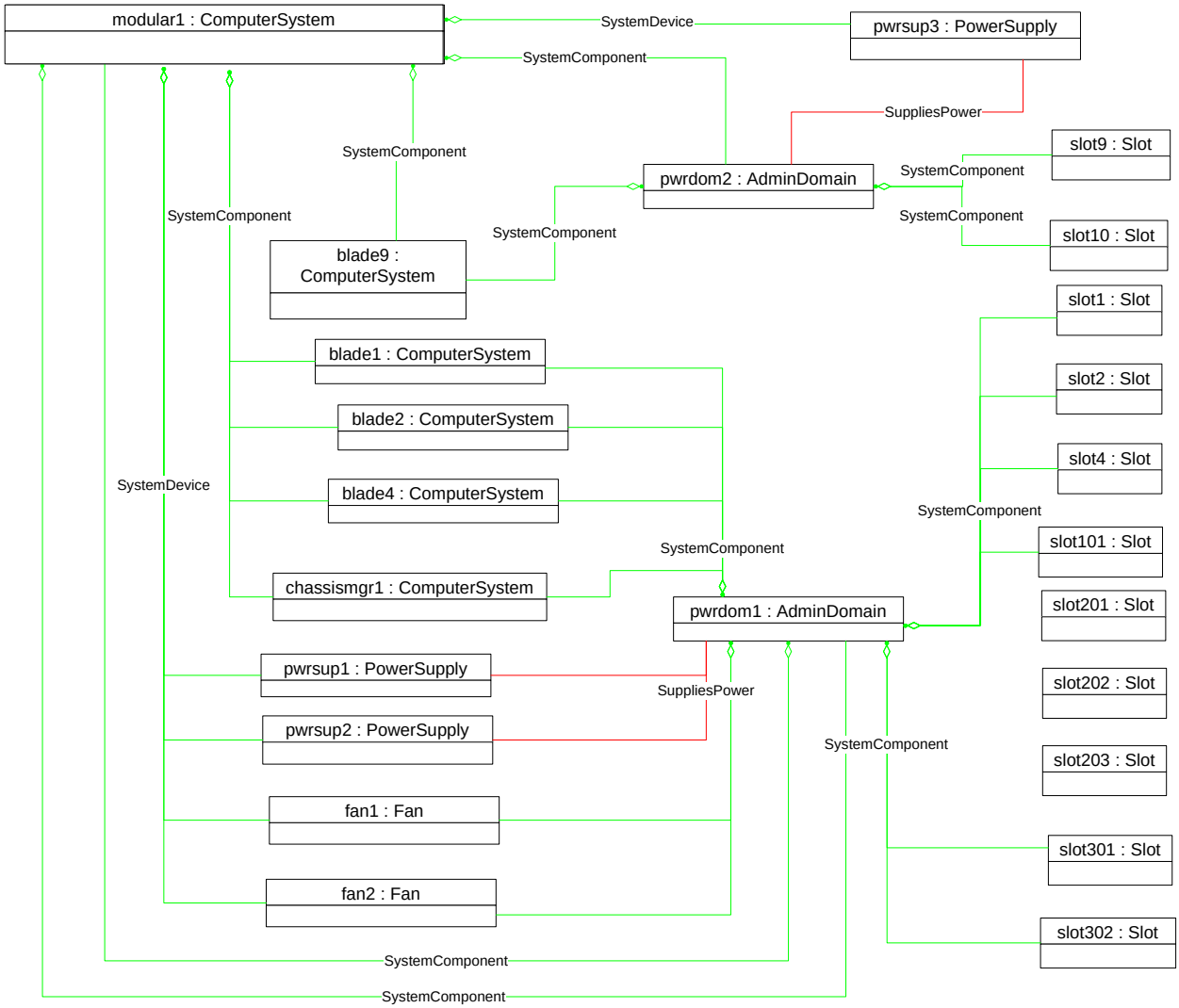


Figure 3 – Chassis Capacity and Compatibility

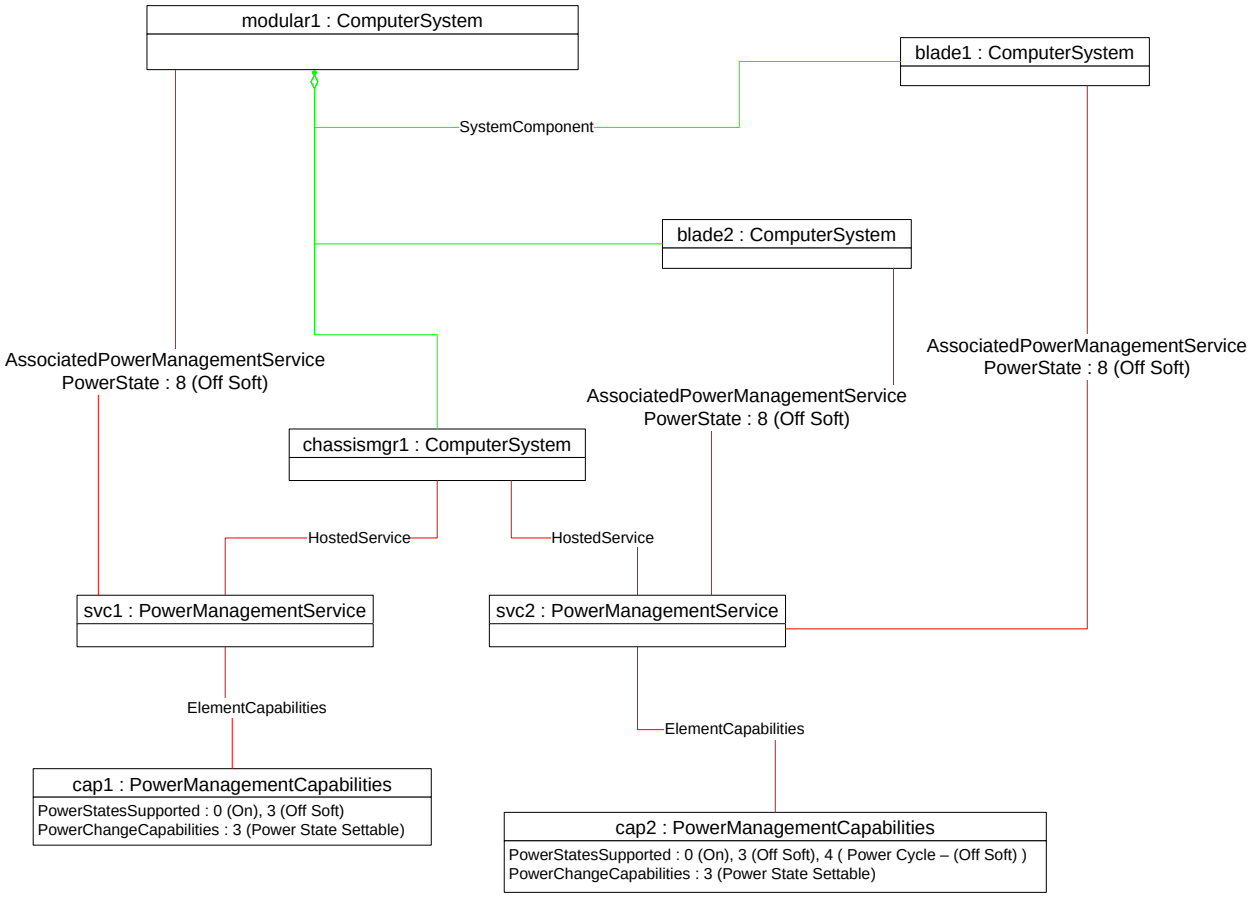
Figure 4 illustrates the modeling of power domains in the modular system. Two power domains are modeled, pwrdom1 and pwrdom2. Components in pwrdom1 receive power from power supplies pwrsup1 and pwrsup2, which is indicated by the instances of CIM_SuppliesPower that associate pwrsup1 and pwrsup2 with pwrdom1. Components in pwrdom2 receive power from pwrsup3. 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**

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

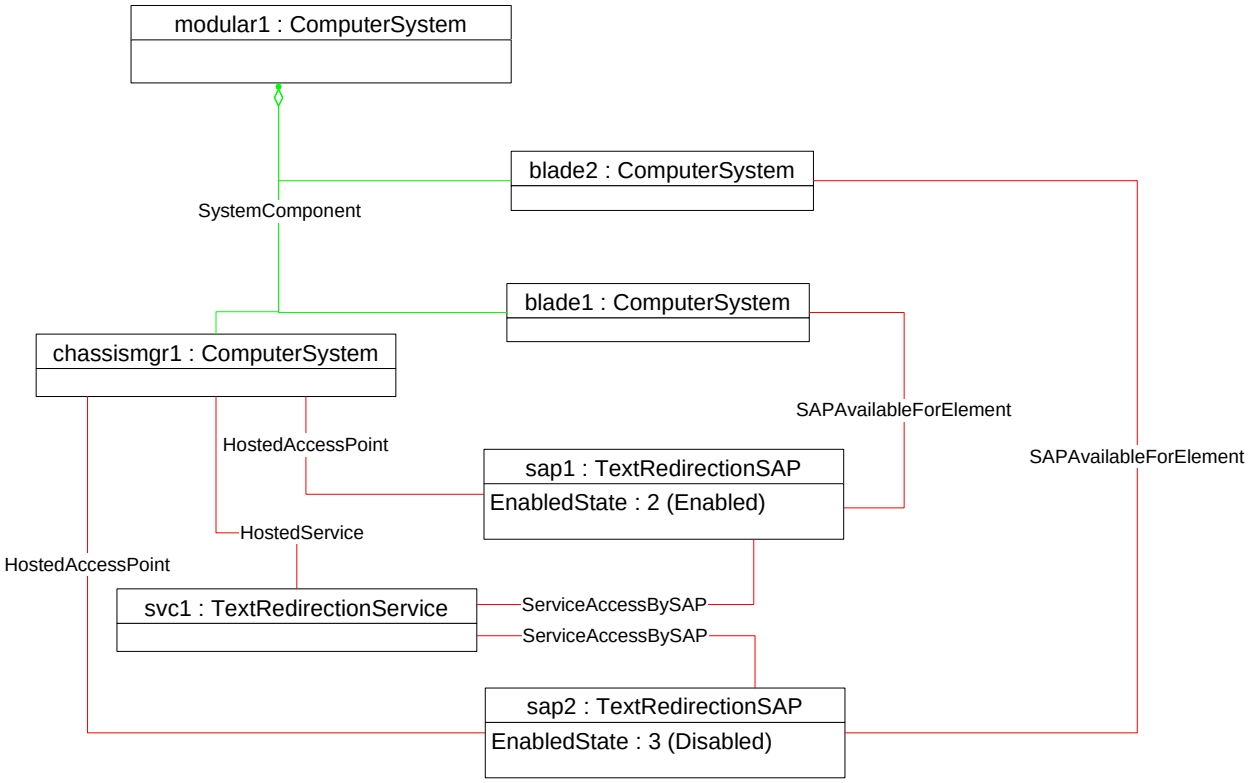


Figure 6 – Text Console Redirection Hosted on Chassis Manager

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

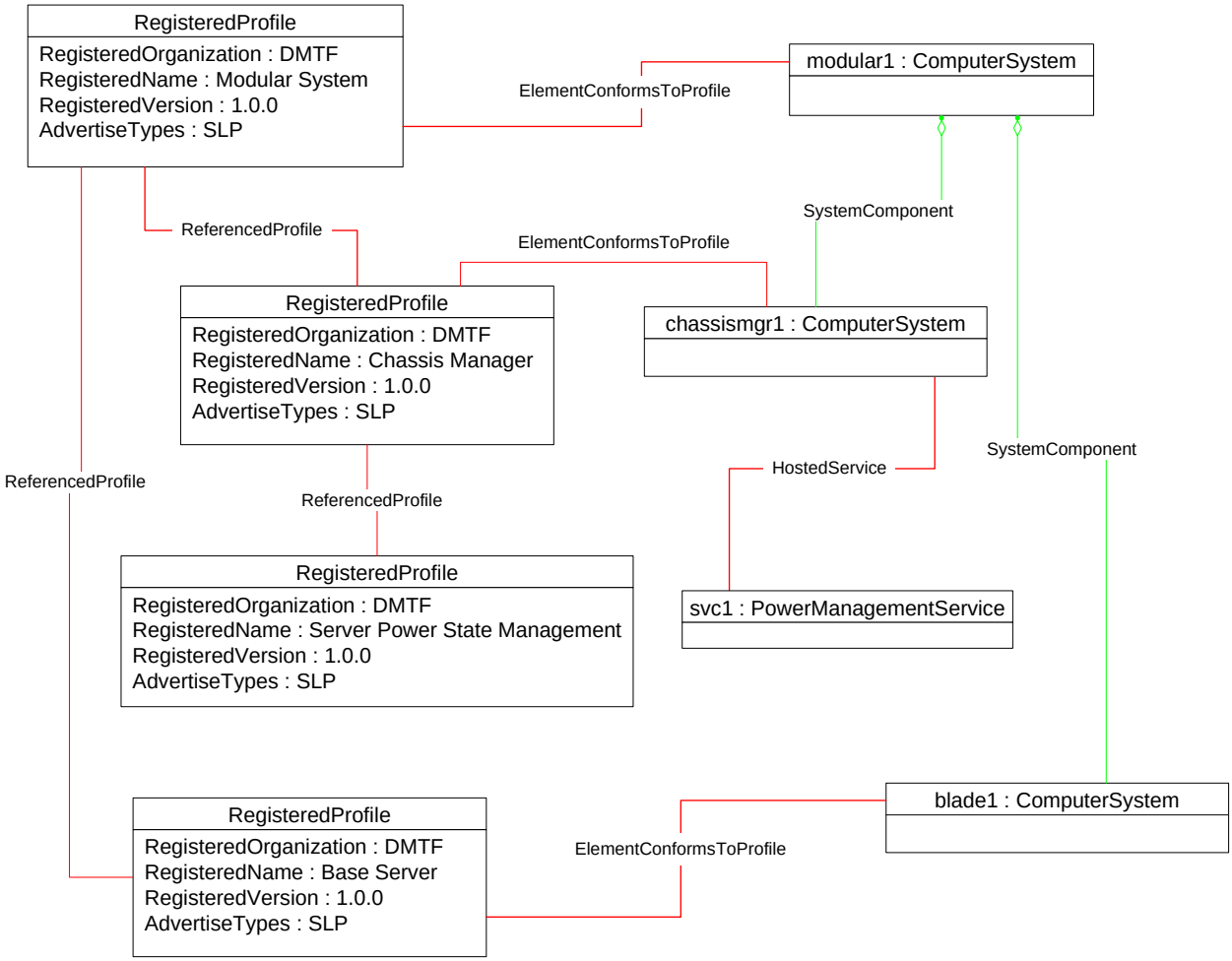


Figure 7 – Registered Profile

9.2 Find the CIM_ComputerSystem Instance for the Enclosure

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

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

9.3 Query Chassis Capacity

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

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

9.4 Query Chassis Component Presence

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

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

9.5 Query Chassis Manager Presence

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

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

9.6 Find All Power Domains for the Modular System

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

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

9.7 Determine the Power Supply for a Component

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

1. Query for an instance of CIM_SuppliesPower that references the CIM_ManagedSystemElement instance.

    1.1 If one or more such instances exist, the associated instances of CIM_PowerSupply supply power to the CIM_ManagedSystemElement instance.

2. Query for an instance of CIM_SystemComponent that references the CIM_ManagedSystemElement instance.

    2.1 If the GroupComponent reference is to an instance of CIM_AdminDomain, query for an instance of CIM_SuppliesPower that references the CIM_AdminDomain instance.

    2.2 If one or more such instances exist, the associated instances of CIM_PowerSupply supply power to the CIM_ManagedSystemElement instance.

3. Query for an instance of CIM_SystemComponent (or a subclass) in which the CIM_ManagedSystemElement instance is the value of the PartComponent reference and an instance of CIM_ComputerSystem is the value of the GroupComponent reference.

    3.1 Find all instances of CIM_PowerSupply that are associated with the CIM_ComputerSystem instance through the CIM_SystemDevice association.

    3.2 If one or more such instances exist, the associated instances of CIM_PowerSupply supply power to the CIM_ManagedSystemElement instance.

4. If the instance of CIM_ManagedSystemElement is an instance of CIM_Slot, complete the following steps:

    4.1 Follow the CIM_Container or CIM_PackageInSlot associations to an instance of CIM_PhysicalElement that represents an outer container.

        4.1.1 If the instance of CIM_PhysicalElement is an instance of CIM_PhysicalPackage or a subclass, query for an instance of CIM_ComputerSystemPackage that references the CIM_PhysicalPackage instance. If not, repeat step 4.1.

        4.1.2 If such an instance exists, select the CIM_ComputerSystem instance and proceed to step 4.2. If not, repeat step 4.1.

    4.2 Find all instances of CIM_PowerSupply that are associated with the CIM_ComputerSystem instance through the CIM_SystemDevice association.

    4.3 If one or more such instances exist, the associated instances of CIM_PowerSupply supply power to the CIM_ManagedSystemElement instance.

9.8 Find All Cooling Domains for the Modular System

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

1) Find instances of CIM_AdminDomain that are associated with the Central Instance through an instance of CIM_SystemComponent whose PartComponent property references the CIM_AdminDomain instance.

2) For each instance of CIM_AdminDomain, determine if the ElementName property matches "Cooling Domain".

9.9 Determine the Fan for a Component

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

1. Query for an instance of CIM_AssociatedCooling that references the CIM_ManagedSystemElement instance.

    1.1 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to the CIM_ManagedSystemElement.

2. Query for an instance of CIM_SystemComponent that references the CIM_ManagedSystemElement instance.

    2.1 If the GroupComponent reference is to an instance of CIM_AdminDomain, query for an instance of CIM_AssociatedCooling that references the CIM_AdminDomain instance.

    2.2 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to the CIM_ManagedSystemElement.

3. Query for an instance of CIM_SystemComponent (or a subclass) in which the CIM_ManagedSystemElement instance is the value of the PartComponent reference and an instance of CIM_ComputerSystem is the value of the GroupComponent reference.

    3.1 Find all instances of CIM_Fan that are associated with the CIM_ComputerSystem instance through the CIM_SystemDevice association.

    3.2 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to the CIM_ManagedSystemElement.

4. If the instance of CIM_ManagedSystemElement is an instance of CIM_Slot, complete the following steps:

    4.1 Follow the CIM_Container or CIM_PackageInSlot associations to an instance of CIM_PhysicalElement that represents an outer container.

        4.1.1 If the instance of CIM_PhysicalElement is an instance of CIM_PhysicalPackage or a subclass, query for an instance of CIM_ComputerSystemPackage that references the CIM_PhysicalPackage instance. If not, repeat step 4.1.

            4.1.1.1 If such an instance exists, select the CIM_ComputerSystem instance and proceed to step 4.2. If not, repeat step 4.1.

    4.2 Find all instances of CIM_Fan that are associated with the CIM_ComputerSystem instance through the CIM_SystemDevice association.

    4.3 If one or more such instances exist, the associated instances of CIM_Fan provide cooling to the CIM_ManagedSystemElement instance.

10 CIM Elements

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

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		

10.1 CIM_AdminDomain—Power Domain

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

Table 5 – Class: CIM_AdminDomain – Power Domain

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

10.2 CIM_AdminDomain—Cooling Domain

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

Table 6 – Class: CIM_AdminDomain – Cooling Domain

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

10.3 CIM_Chassis

CIM_Chassis is defined by the *Physical Asset Profile*. The requirements denoted in Table 7 are in addition to those mandated by the *Physical Asset Profile*.

Table 7 – Class: CIM_Chassis

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

10.4 CIM_ComputerSystem

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

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

10.5 CIM_ComputerSystemPackage

CIM_ComputerSystemPackage associates the CIM_Chassis instance for the modular enclosure with the CIM_ComputerSystem instance for the modular enclosure. Requirements specified in Table 9 are in addition to those specified in the *Physical Asset Profile*.

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

10.6 CIM_ConcreteDependency

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

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 *

10.7 CIM_PhysicalPackage

CIM_PhysicalPackage is defined by the *Physical Asset Profile*. The requirements denoted in Table 11 are in addition to those mandated by the *Physical Asset Profile*.

Table 11 – Class: CIM_PhysicalPackage

Elements	Requirement	Notes
PackageType	Mandatory	See 7.3.1.

10.8 CIM_RegisteredProfile

CIM_RegisteredProfile identifies the *Modular System Profile* in order for a client to determine whether an instance of CIM_ComputerSystem is conformant with this profile. CIM_RegisteredProfile is defined by the *Profile Registration Profile*. With the exception of the mandatory values specified for the properties in Table 12, the behavior of the CIM_RegisteredProfile instance is in accordance with the *Profile Registration Profile*.

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).

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

10.9 CIM_SystemComponent—Cooling Domains

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

Table 13 – Class: CIM_SystemComponent – Cooling Domains

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

10.10 CIM_SystemComponent—Power Domains

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

Table 14 – Class: CIM_SystemComponent – Power Domains

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

10.11 CIM_SystemComponent—Chassis Manager

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

Table 15 – Class: CIM_SystemComponent – Chassis Manager

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

10.12 CIM_SystemComponent—Processor Blades

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

Table 16 – Class: CIM_SystemComponent – Processor Blades

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

ANNEX A
(informative)

Change Log

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