Document Number: DSP1011

Date: 2016-05-17

Version: 1.0.3

Physical Asset Profile

Supersedes: 1.0.2

Document Class: Normative

Document Status: Published

Document Language: en-US

CONTENTS

Figures

Tables

Foreword

The *Physical Asset Profile* (DSP1011) was prepared by the Physical Platform Profiles Working Group and Server Management Working Group.

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

Acknowledgments

The DMTF acknowledges the following individuals for their contributions to this document:

Editors:

- Jon Hass – Dell Inc.
- Jeff Hilland – Hewlett Packard Enterprise
- Khachatur Papanyan – Dell Inc.
- Hemal Shah – Broadcom Corporation

Contributors:

- Jon Hass – Dell Inc.
- Jeff Hilland – Hewlett Packard Enterprise
- Arvind Kumar – Intel
- John Leung – Intel
- Jeff Lynch – IBM
- Aaron Merkin – IBM
- Khachatur Papanyan – Dell Inc.
- Hemal Shah – Broadcom Corporation
- Christina Shaw – Hewlett Packard Enterprise
- Perry Vincent – Intel

Introduction

This document describes the physical aspects of the logical elements that the implementation is instantiating. Physical aspects include asset, inventory, and other descriptive physical information. Also included are descriptions of association classes that describe the relationship of physical elements and DMTF profile registration information. 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 must be instantiated and manipulated to represent and manage classes representing physical elements of systems and subsystems 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 representing the component described in this document.

Physical Asset Profile

1 Scope

The *Physical Asset Profile* extends the management capability of the referencing profiles by adding the capability to describe the physical aspects of logical elements that the implementation is instantiating. The profile also describes the relationship between the physical elements and the profile's registration for the schema implementation and version information.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated or versioned references, only the edition cited (including any corrigenda or DMTF update versions) applies. For references without a date or version, the latest published edition of the referenced document (including any corrigenda or DMTF update versions) applies.

2.1 Approved references

DMTF DSP0004, C*IM Infrastructure Specification 2.3*,
http://www.dmtf.org/standards/published_documents/DSP0004_2.3.pdf

DMTF DSP0200, *CIM Operations over HTTP 1.2*,
http://www.dmtf.org/sites/default/files/standards/documents/DSP200.html

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

DMTF DSP1013, *Fan Profile 1.0,*
http://www.dmtf.org/standards/published_documents/DSP1013_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

In this document, some terms have a specific meaning beyond the normal English meaning. Those terms are defined in this clause.

The terms "shall" ("required"), "shall not", "should" ("recommended"), "should not" ("not recommended"), "may", "need not" ("not required"), "can" and "cannot" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Annex H. The terms in parenthesis are alternatives for the preceding term, for use in exceptional cases when the preceding term cannot be used for linguistic reasons. Note that ISO/IEC Directives, Part 2, Annex H specifies additional alternatives. Occurrences of such additional alternatives shall be interpreted in their normal English meaning.

The terms "clause", "subclause", "paragraph", and "annex" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Clause 5.

The terms "normative" and "informative" in this document are to be interpreted as described in ISO/IEC Directives, Part 2, Clause 3. In this document, clauses, subclauses, or annexes labeled "(informative)" do not contain normative content. Notes and examples are always informative elements.

The terms defined in DSP0004, DSP0223, and DSP1001 apply to this document. The following additional terms are used in this document.

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 in order to conform to the document when the specified conditions are met

3.4

mandatory

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

3.10

shall not

indicates requirements 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

Delimited Substring

a substring element of the VendorCompatibilityStrings property of a Physical Element or an instance of CIM_ConfigurationCapacity. The substring starts at the beginning of the string (representing an element in the array of the VendorCompatibilityStrings property) and terminates at the end of the string, or at a character that precedes a colon (:).

3.15

Physical Element

an instance of a CIM_PhysicalElement subclass (such as CIM_PhysicalConnector, CIM_Slot, CIM_PhysicalComponent, CIM_Chip, CIM_PhysicalMemory, CIM_PhysicalPackage, CIM_PhysicalFrame, CIM_Chassis, CIM_Rack, and CIM_Card) that represents a physical element

3.16

Physical Package

an instance of a CIM_PhysicalPackage or a CIM_PhysicalPackage subclass (such as CIM_PhysicalFrame, CIM_Chassis, CIM_Rack, and CIM_Card) or CIM_PhysicalComponent or CIM_PhysicalComponent subclass (such as CIM_Chip or CIM_PhysicalMemory) that represents a package

3.17

System Chassis

an instance of the CIM_PhysicalElement or CIM_Chassis that is associated to an instance of CIM_System or CIM_ComputerSystem through the CIM_SystemPackaging or CIM_ComputerSystemPackage association, representing the physical package of the managed system.

4 Symbols and abbreviated terms

The abbreviations defined in DSP0004, DSP0223, and DSP1001 apply to this document. The following additional abbreviations are used in this document.

4.1

CIM

Common Information Model

4.2

FRU

Field Replaceable Unit

5 Synopsis

Profile Name: Physical Asset

Version: 1.0.3

Organization: DMTF

CIM Schema version: 2.22

Central Class: CIM_PhysicalElement

Scoping Class: CIM_ManagedSystemElement

The *Physical Asset Profile* extends the management capability of the referencing profiles by adding the capability to describe the physical aspects of the logical elements that the implementation is instantiating. Physical aspects include asset, inventory, and other descriptive physical information.

Table 1 – Referenced profiles

Profile Name	Organization	Version	Relationship	Behavior
Profile Registration	DMTF	1.0	Mandatory	

The Central Instance for the *Physical Asset Profile* shall be the instance of the CIM_PhysicalElement subclass.

The Scoping Instance for the *Physical Asset Profile* shall be the instance of the CIM_ManagedSystemElement. Note that this may include the subclass of CIM_System, the CIM_ComputerSystem class. The Scoping Instance is determined using the algorithm described in clause 7.2.

6 Description

The *Physical Asset Profile* describes the necessary elements needed to provide the descriptive and asset information about the physical components in a managed domain and their topology. The profile does not cover the geographic location of the physical assets.

Figure 1 shows the CIM classes that are used in this profile. (For simplicity, the prefix CIM_ has been removed from the names of the classes.) A Physical Element (see clause 3.15) describes the physical properties, including the FRU information, of a managed element. The capabilities of the Physical Elements are described by the properties of the CIM_PhysicalAssetCapabilities class. The Physical Elements could be associated to the logical representation of the managed element through the CIM_Realizes association. The enclosures or chassis of the managed systems are represented by a CIM_PhysicalElement or CIM_Chassis instance that is associated to the CIM_System/CIM_ComputerSystem instance through the CIM_SystemPackaging/CIM_ComputerSystemPackage association and are referred to as a System Chassis (see clause 3.17). Configuration capacity of the System Chassis is also represented within this profile by CIM_ConfigurationCapacity instances.

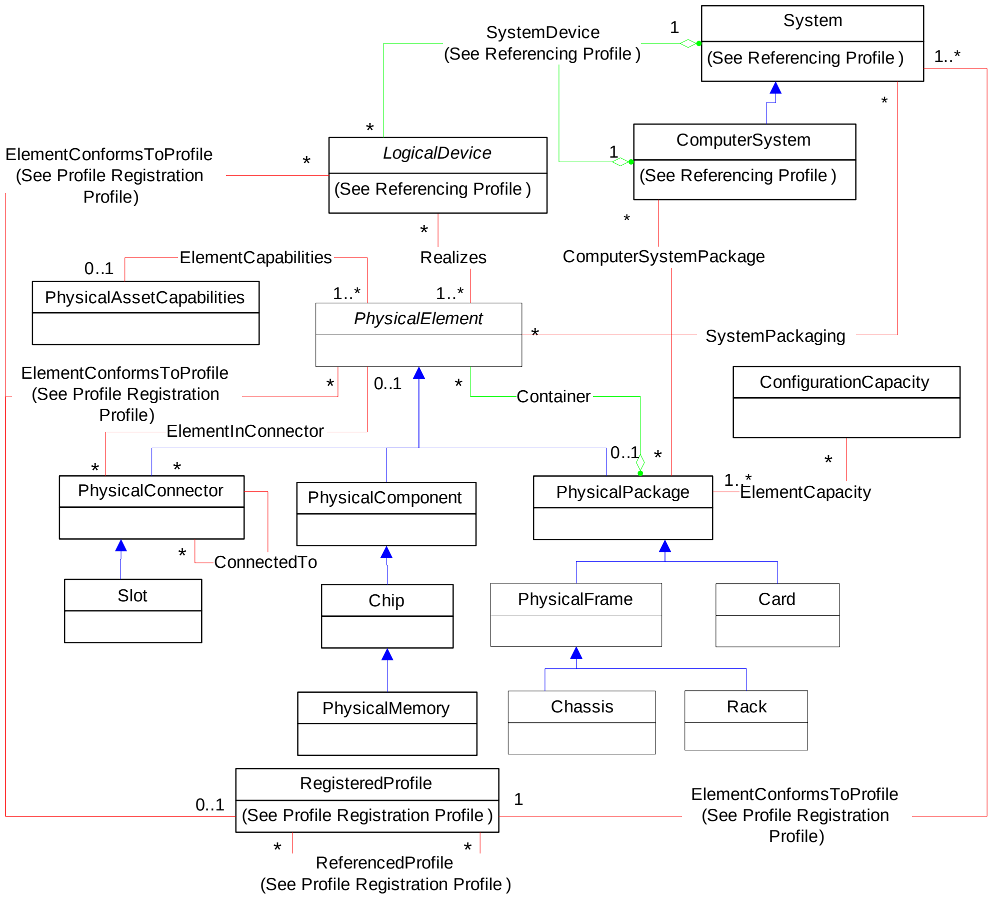


Figure 1 – Physical Asset Profile: Profile class diagram

Physical Elements can be also arranged in a topology. The CIM_Container, CIM_ConnectedTo, and CIM_ElementInConnector associations are used to associate the Physical Elements and create the physical topology of the managed elements.

Figure 1 also represents the ecosystem of *Physical Asset Profile* classes, illustrating their relationship with classes of referencing profiles. The referencing profiles can identify the subclass of CIM_PhysicalElement to be used for representing the physical aspects of the managed element. For example, the referencing profiles that contain a CIM_LogicalDevice subclass can restrict the associated subclass of CIM_PhysicalPackage to CIM_PhysicalMemory for instantiation of the *Physical Asset Profile*. Such restrictions will be described in the referencing profiles.

The *Physical Asset Profile* is advertised through the CIM_RegisteredProfile instance.

The *Physical Asset Profile* can be instantiated to represent a combination of the following scenarios:

- the physical aspects of a managed system, such as the FRU information for the chassis (see clause 7.6)
- the physical aspects of a specific managed element, such as the FRU information of a fan (see clause 7.3)

- the physical hierarchy of a managed system, such as the relationship between chassis, slots, and packages (see clause 7.8)
- the configuration capacity of a managed element, such as the minimum and maximum number of certain types of packages that the managed system can handle (see clause 7.7)

7 Implementation

This clause details the requirements related to the arrangement of instances and their properties for implementations of this profile.

7.1 Physical element

The implementation shall instantiate at least one instance of the subclass of CIM_PhysicalElement (Physical Element). Referencing profiles may state the subclass of CIM_PhysicalElement that is to be instantiated as part of the *Physical Asset Profile*.

At least one instance of CIM_Realizes, CIM_ComputerSystemPackage, or CIM_SystemPackaging association class shall reference an instance of a subclass of CIM_PhysicalElement (Physical Element).

Every Physical Element shall be referenced by at least one of the following properties: CIM_ComputerSystemPackage.Antecedent, CIM_SystemPackaging.Antecedent, CIM_Realizes.Antecedent, CIM_Container.PartComponent, or CIM_ElementInConnector.Dependent.

7.2 Finding the Scoping Instance of the CIM_System or CIM_ComputerSystem class

The following algorithm shall be used for locating the Scoping Instance of the CIM_System or CIM_ComputerSystem class starting from any selected Physical Element.

I. If the selected instance is of a Physical Package, proceed as follows:

A. If the Physical Package is associated to the CIM_LogicalDevice through the CIM_Realizes association or to the CIM_System/CIM_ComputerSystem through the CIM_SystemPackaging/CIM_ComputerSystemPackage association, the Scoping Instance of the *Physical Asset Profile* shall be either of the following:

a. the Scoping Instance of the CIM_LogicalDevice instance that is associated to the Physical Package through the instance of CIM_Realizes

b. the Scoping Instance of CIM_System/CIM_ComputerSystem instance that is associated to the Physical Package through the instance of CIM_SystemPackaging or CIM_ComputerSystemPackage

B. If the Physical Package is not associated to the CIM_LogicalDevice through the CIM_Realizes association or to the CIM_System or CIM_ComputerSystem through the CIM_SystemPackaging or CIM_ComputerSystemPackage association, proceed as follows:

1. If the Physical Package is the Dependent or PartComponent reference in CIM_ElementInConnector or CIM_Container associations, respectively, choose one of the following paths:

a. If the Antecedent or GroupComponent reference of the association is a Physical Package, select the Antecedent or GroupComponent referenced instance, and go to I.A.

b. Else if the Antecedent or GroupComponent reference of the association is a Physical Element:

(1) If the Physical Element is associated to the CIM_LogicalDevice through the CIM_Realizes association, the Scoping Instance of the *Physical Asset Profile* shall be the Scoping Instance of the CIM_LogicalDevice instance.

(2) If the Physical Element instance is not associated to the CIM_LogicalDevice through the CIM_Realizes association:

(a) If the Physical Element is the PartComponent reference in the CIM_Container association:

1) If a Physical Package is the GroupComponent reference for the CIM_Container association, select the GroupComponent referenced instance, and go to I.A.

2) If a Physical Element is the GroupComponent or Antecedent reference, go to I.B.1.b(1).

(b) If the Physical Element is not the PartComponent or Dependent reference in a CIM_Container association, the Scoping Instance shall be the Central Instance; thus, the Central Instance is associated to the CIM_RegisteredProfile instance.

2. Else the Scoping Instance shall be the Central Instance, thus, the Central Instance is associated to the CIM_RegisteredProfile instance.

II. If the instance is not a Physical Package, go to I.B.1.b(1).

7.3 Modeling the physical aspects of logical representation of devices

The implementation may implement the physical aspects of a managed device through instantiation of a Physical Element.

When the physical aspects of the logical device are implemented, the CIM_LogicalDevice subclass instance, which represents the logical device, shall be associated with the Physical Element, which represents the physical aspects of the logical device, through the CIM_Realizes association.

7.4 Support for the Physical Element’s FRU information

The Physical Element’s support of FRU information shall be advertised by a CIM_PhysicalAssetCapabilities instance associated with the Physical Element. At most, one instance of CIM_PhysicalAssetCapabilities shall be associated with the Physical Element through the CIM_ElementCapabilities association.

When no CIM_PhysicalAssetCapabilities instance is associated to the Physical Element, the Physical Element’s FRU information may not be supported.

When a CIM_PhysicalAssetCapabilities instance is associated to the Physical Element and the CIM_PhysicalAssetCapabilities.FRUInfoSupported has a value of TRUE, the Physical Element’s FRU information shall be supported.

When FRU information is supported, the implementation shall populate the properties of the Physical Element below with non-null, non-blank values. At least one of these properties shall be non-null, non-blank without any whitespace characters (e.g., pattern “[^WSP]+”). Any property stated below shall not be populated with a string containing only whitespace characters. If the SKU property is non-null, it shall be used to convey the FRU number. Some combination of the properties below should be used for replacement part information.

- Manufacturer
- Model
- PartNumber

- SerialNumber
- SKU

7.5 Compatibility of Physical Packages

When the Physical Package is instantiated, the implementation may represent the compatibility of the Physical Package. In that case, the conditions and requirements in this clause shall apply.

The compatibility between the physical packages, which are represented by Physical Packages, and slots, which are represented by CIM_Slot instances, shall be advertised through the VendorCompatibilityStrings property.

The VendorCompatibilityStrings property of a Physical Package and an instance of CIM_Slot shall be an array of strings, each uniquely identifying the specific type of package and matching a ":" character-free, non-zero length string, delimited by ":"character (pattern "[^:]+(:[^:]+)+").

Only if the physical package represented by the Physical Package can be inserted into the slot represented by the instance of CIM_Slot, the VendorCompatibilityStrings property of Physical Package shall contain an element with a Delimited Substring equal to a string of one of the elements from the VendorCompatibilityStrings property of an instance of CIM_Slot.

7.6 Modeling System Chassis

The implementation may instantiate a System Chassis. When a System Chassis is instantiated, the System Chassis shall be associated with the instance of CIM_System through the instance of CIM_SystemPackaging, or with the instance of CIM_ComputerSystem through the instance of CIM_ComputerSystemPackage.

7.7 Modeling configuration capacity

The implementation may advertise the configuration capacity of the physical packages within the chassis, including the chassis itself. The configuration capacity shall be represented through the CIM_ConfigurationCapacity instances.

When a System Chassis is present, the instrumentation shall associate all the instances of CIM_ConfigurationCapacity to the System Chassis through the instances of CIM_ElementCapacity. Additionally, when the configuration capacity is for a particular physical package represented by a Physical Package, the instrumentation may associate the Physical Package with the CIM_ConfigurationCapacity through an instance of CIM_ElementCapacity.

When instances of CIM Slot are instantiated, for each unique value of the CIM_Slot.VendorCompatibilityStrings, an instance of CIM_ConfigurationCapacity with an equal value for the CIM_ConfigurationCapacity.VendorCompatibilityStrings property shall exist. Additional instances of CIM_ConfigurationCapacity may exist.

When CIM_Slot instances are not instantiated or the CIM_Slot.VendorCompatibilityStrings property is not instrumented, the CIM_ConfigurationCapacity.VendorCompatibilityStrings array property shall contain an element with a Delimited Substring that is equal to a string of one of the elements from the VendorCompatibilityStrings array property of a Physical Package that can be part of the configuration.

7.8 Modeling physical hierarchy

The physical hierarchy is represented by relationship and containment of Physical Elements. The implementation may represent the physical hierarchy as follows:

- When a physical element resides within a package, the Physical Element shall be associated with the Physical Package through the CIM_Container association.
- When a package is plugged or connected to a slot or connector, the Physical Package shall be associated with the CIM_PhysicalConnector or CIM_Slot instance through the CIM_ElementInConnector association.
- When physical connectors or slots are connected, the CIM_PhysicalConnector or CIM_Slot instances shall be associated through the CIM_ConnectedTo association.

7.9 Modeling a physical memory

The implementation may implement the physical aspects of a memory inside the system through instantiation of the CIM_PhysicalMemory class.

When a physical memory is modeled as an instance of CIM_PhysicalMemory, the CIM_PhysicalMemory.Speed property represents the speed of the physical memory in nanoseconds. The following requirements apply for CIM_PhysicalMemory.Speed:

- If the speed of the physical memory is less than one nanosecond or unknown, then the CIM_PhysicalMemory.Speed property shall be set to 0.
- If the speed of the physical memory is variable, then the CIM_PhysicalMemory.Speed property shall be set to 2^32-1 (nanoseconds).

8 Methods

This clause details the requirements for supporting intrinsic operations for the CIM elements defined by this profile. The *Physical Asset Profile* does not define any extrinsic methods.

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_Card

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_Chassis

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_Chip

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.

8.5 CIM_ComputerSystemPackage

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_ComputerSystemPackage

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

8.6 CIM_ConfigurationCapacity

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.

8.7 CIM_ConnectedTo

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_ConnectedTo

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

8.8 CIM_Container

Table 4 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 4, 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 4 – Operations: CIM_Container

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

8.9 CIM_ElementCapabilities

Table 5 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 5, 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 5 – Operations: CIM_ElementCapabilities

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

8.10 CIM_ElementCapacity

Table 6 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 6, 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 6 – Operations: CIM_ElementCapacity

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

8.11 CIM_ElementInConnector

Table 7 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 7, 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 7 – Operations: CIM_ElementInConnector

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

8.12 CIM_PhysicalAssetCapabilities

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.

8.13 CIM_PhysicalComponent

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.

8.14 CIM_PhysicalConnector

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.

8.15 CIM_PhysicalFrame

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.

8.16 CIM_PhysicalMemory

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.

8.17 CIM_PhysicalPackage

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.

8.18 CIM_Rack

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.

8.19 CIM_Realizes

Table 8 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 8, 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 8 – Operations: CIM_Realizes

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

8.20 CIM_Slot

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.

8.21 CIM_SystemPackaging

Table 9 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 9, 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 9 – Operations: CIM_SystemPackaging

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

9 Use cases

9.1 System chassis FRU information

Figure 2 represents a possible instantiation of the *Physical Asset Profile*. In this case, the physical aspects of the instance of CIM_ComputerSystem are represented by an instance of CIM_Chassis through a CIM_ComputerSystemPackage association. The Tag property of Chassis1 represents the asset tag of the chassis. The TRUE value of the FRUInfoSupported property of capabilities1 indicates that chassis1 contains non-zero, non-blank properties describing FRU information such as PartNumber, SerialNumber, Model, and Manufacturer. (See clause 7.4 for more details.) Profile2 advertises the implemented *Physical Asset Profile* information.

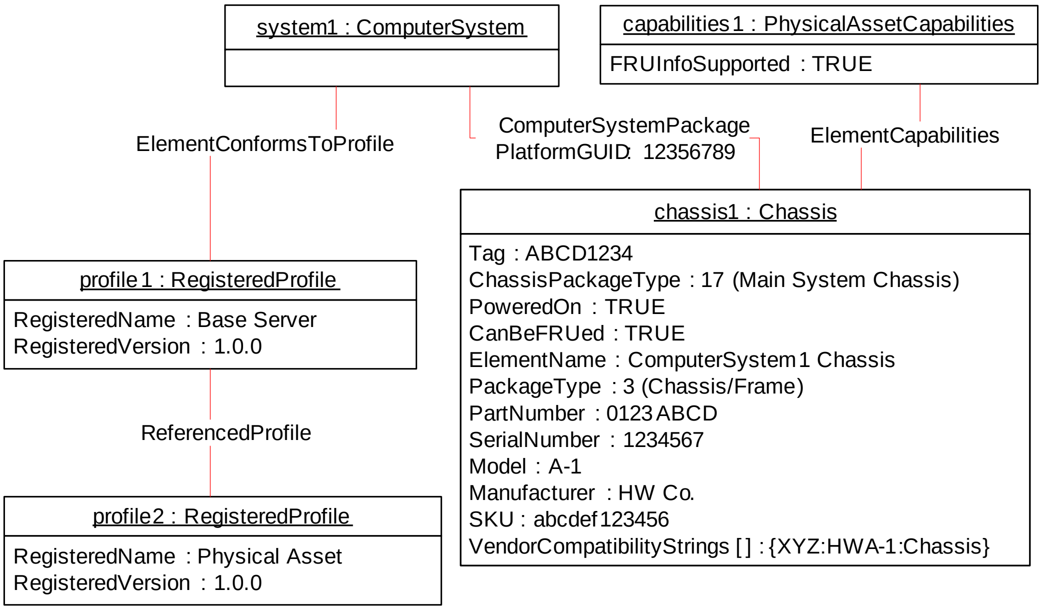

system1 : ComputerSystem
capabilities1 : PhysicalAssetCapabilities
FRUInfoSupported : TRUE
ElementConformsToProfile
ComputerSystemPackage
PlatformGUID: 12356789
ElementCapabilities
chassis1 : Chassis
Tag : ABCD1234
ChassisPackageType : 17 (Main System Chassis)
PoweredOn : TRUE
CanBeFRUed : TRUE
ElementName : ComputerSystem1 Chassis
PackageType : 3 (Chassis/Frame)
PartNumber : 0123ABCD
SerialNumber : 1234567
Model : A-1
Manufacturer : HW Co.
SKU : abcdef123456
VendorCompatibilityStrings [] : {XYZ:HWA-1:Chassis}
profile1 : RegisteredProfile
RegisteredName : Base Server
RegisteredVersion : 1.0.0
ReferencedProfile
profile2 : RegisteredProfile
RegisteredName : Physical Asset
RegisteredVersion : 1.0.0

Figure 2 – System chassis object diagram

9.2 Fan package FRU information

Figure 3 represents another possible instantiation of the *Physical Asset Profile*. The instance of CIM_PhysicalPackage represents the physical properties of the given instance of CIM_Fan through a CIM_Realizes association. The CIM_PhysicalPackage.Tag property represents the asset tag of the fan1. The TRUE value of the FRUInfoSupported property of capabilities1 indicates that physicalpackage1 contains non-zero, non-blank properties describing FRU information such as PartNumber, SerialNumber, Model, Manufacturer, and SKU. (See clause 7.4 for more details.)

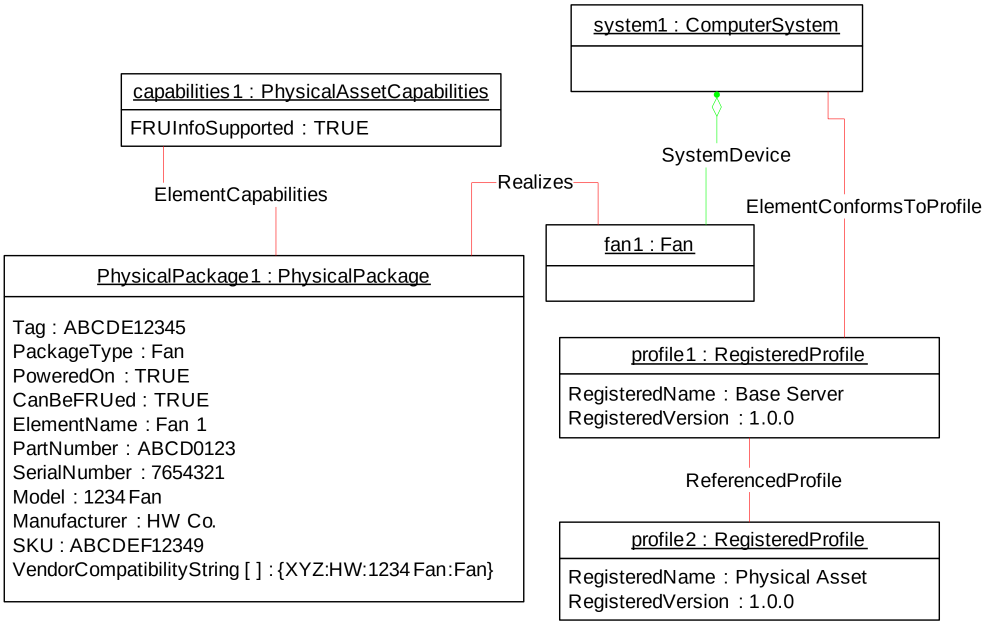


Figure 3 – CIM_PhysicalPackage object diagram

9.3 Finding the Scoping Instance for a fan package

Figure 4 represents another possible instantiation of *Physical Asset Profile*. To find the Scoping Instance of PhysicalPackage1, the client needs to select the fan1 associated through the CIM_Realizes association and then find the Scoping Instance for fan1. As defined in the *Fan Profile*, the Scoping Instance of fan1 is the CIM_ComputerSystem instance associated to fan1 through the CIM_SystemDevice association: system1. Thus, system1 is the Scoping Instance of PhysicalPackage1. By traversing through the CIM_ElementConformsToProfile and subsequently the CIM_ReferencedProfile association, the client can find profile2, which advertises the *Physical Asset Profile* information.

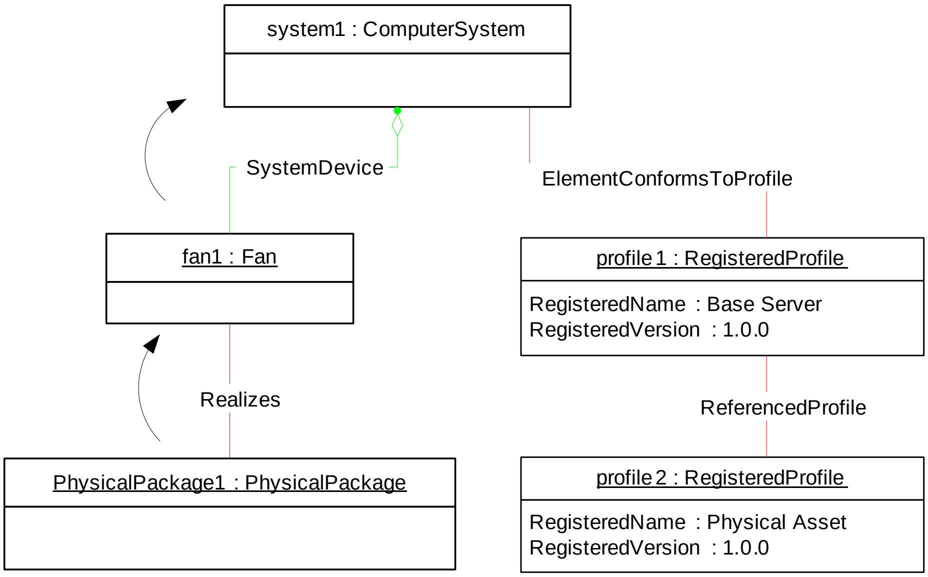
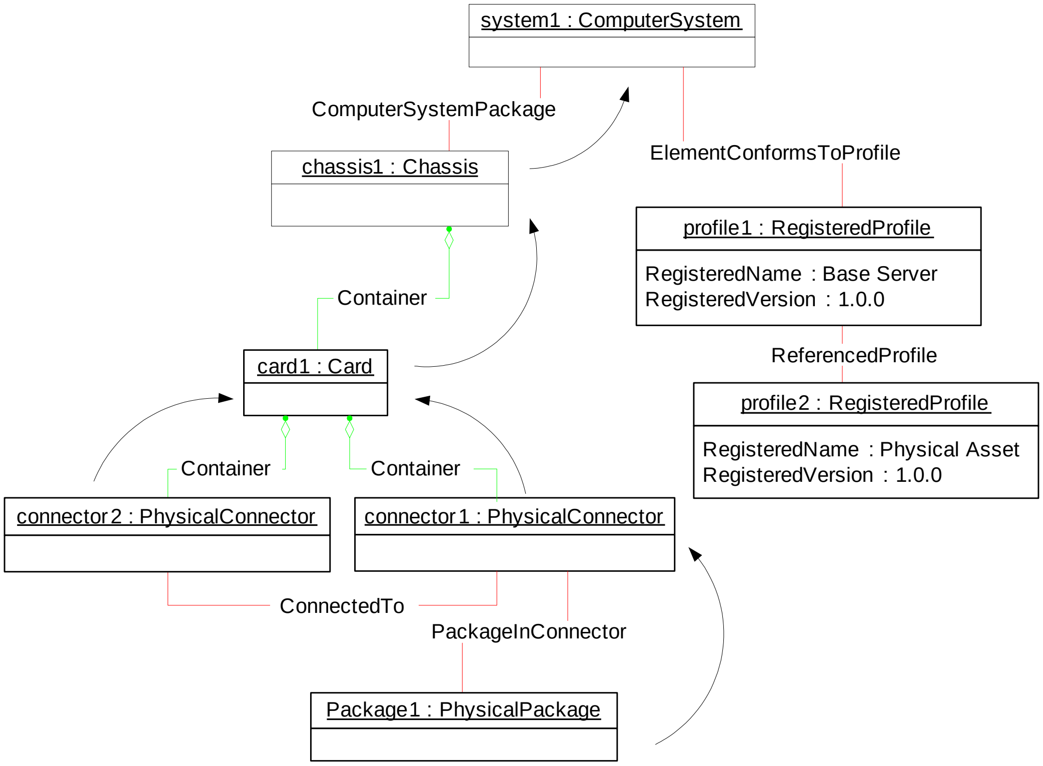

system1 : ComputerSystem
SystemDevice
ElementConformsToProfile
fan1 : Fan
profile1 : RegisteredProfile
RegisteredName : Base Server
RegisteredVersion : 1.0.0
Realizes
ReferencedProfile
PhysicalPackage1 : PhysicalPackage
profile2 : RegisteredProfile
RegisteredName : Physical Asset
RegisteredVersion : 1.0.0

Figure 4 – Scoping Instance: Logical device object diagram

9.4 Physical topology and finding the Scoping Instance

Figure 5 represents another possible instantiation of the *Physical Asset Profile*. To find the Scoping Instance of package1, because package1 is referenced by the CIM_ElementInConnector.Dependent property, the client needs to select connector1, which is referenced by the CIM_ElementInConnector.Antecedent property. Then, because connector1 is referenced by the CIM_Container.PartComponent property, the client needs to select card1, which is referenced by the CIM_Container.GroupComponent. Then, because card1 is referenced by the CIM_Container.PartComponent property, the client needs to select chassis1, which is referenced by the CIM_Container.GroupComponent. Then, because chassis1 is associated to system1 through the CIM_ComputerSystemPackage association, system1 is the Scoping Instance of package1. The client can traverse through the CIM_ElementConformsToProfile and, subsequently, the CIM_ReferencedProfile association, to find profile2, which advertises the *Physical Asset Profile* information.

NOTE To enable finding the Scoping Instance of connector2, the implementation has instantiated an instance of CIM_Container that references card1 and connector2. Merely instantiating the instance of CIM_ConnectedTo referencing connector2 will not conform to the algorithm described in clause 7.2.


system1 : ComputerSystem
ComputerSystemPackage
ElementConformsToProfile
chassis1 : Chassis
profile1 : RegisteredProfile
RegisteredName : Base Server
RegisteredVersion : 1.0.0
Container
card1 : Card
ReferencedProfile
profile2 : RegisteredProfile
RegisteredName : Physical Asset
RegisteredVersion : 1.0.0
Container
Container
connector2 : PhysicalConnector
connector1 : PhysicalConnector
ConnectedTo
PackageInConnector
Package1 : PhysicalPackage

645

646 **Figure 5 – Scoping Instance: Physical topology object diagram**

9.5 Physical topology

Figure 6 represents another possible instantiation of the *Physical Asset Profile*. Chassis1 is a System Chassis of system1. Physicalpackage1 is a Physical Package for fan1. The physical topology of chassis1 contains a single level because card1, slot1, chip1, pmem1, component1, connector1, and physicalpackage1 are all directly associated to chassis1 through the instances of CIM_Container.

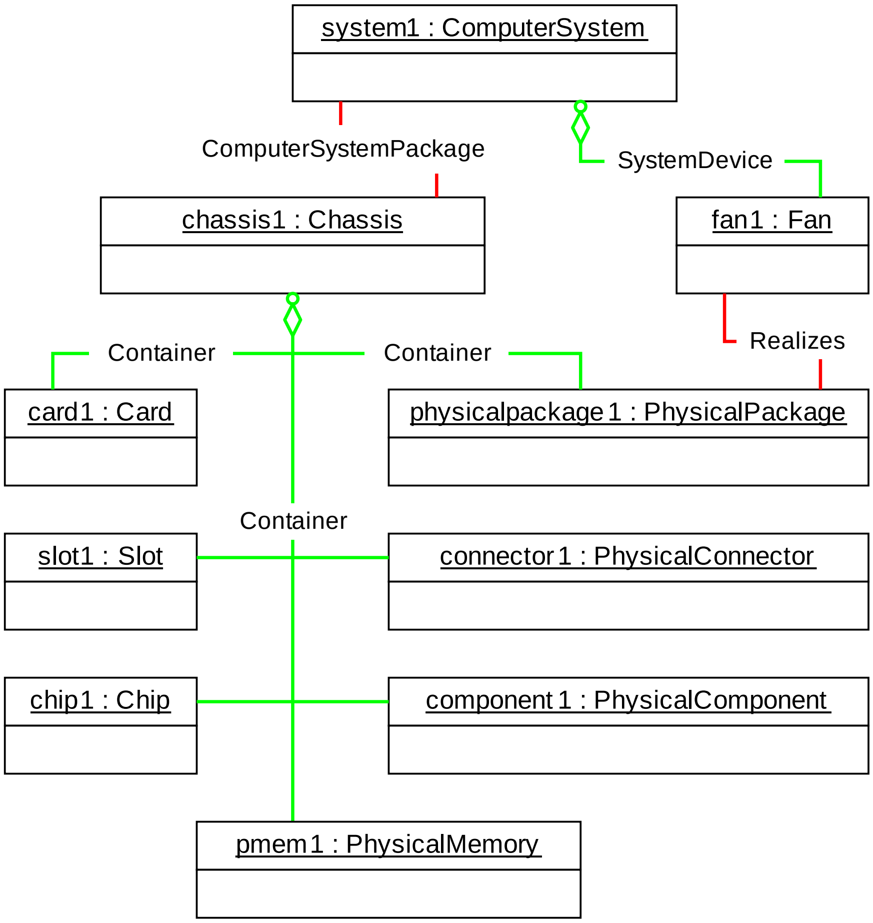


Figure 6 – Physical Asset Profile: Topology object diagram

9.6 Physical memory

Figure 7 represents another possible instantiation of the *Physical Asset Profile*. System1's system memory is represented by Memory1. Memory1's physical aspects are represented by pmem1. chassis1 is a System Chassis of system1. chassis1 contains slot1, into which the memory package, memorypkg1, is plugged. memorypkg1 contains pmem1, the physical representation of the system memory, Memory1.

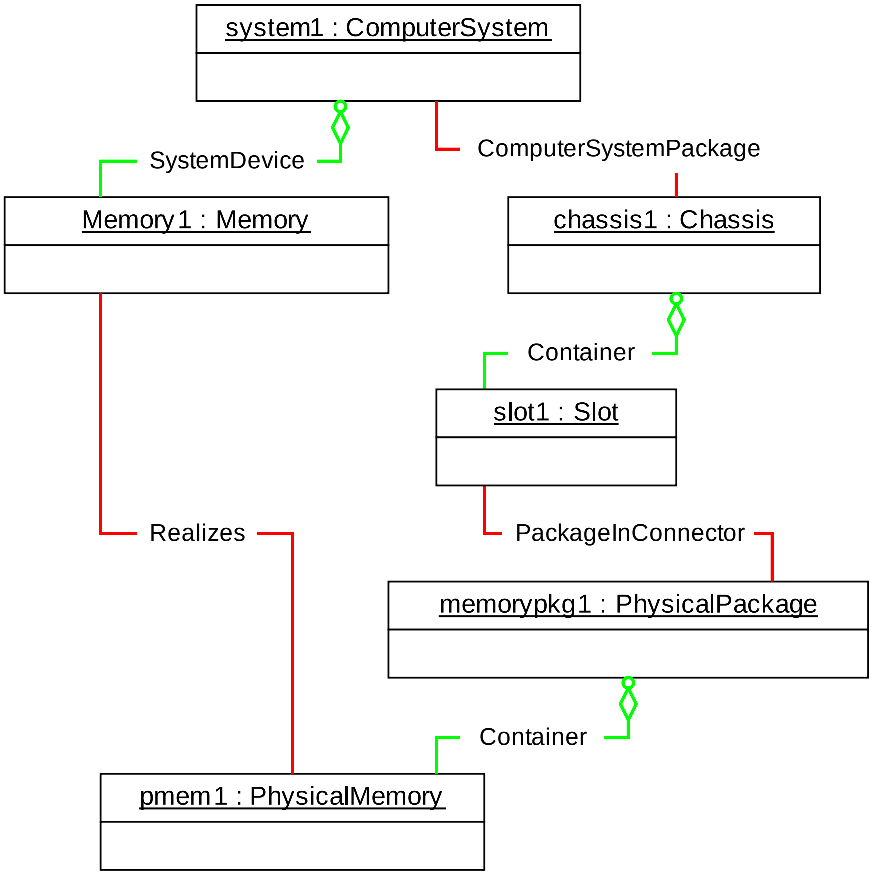


Figure 7 – Physical memory topology object diagram

9.7 Representing configuration capacity

Figure 8 represents another possible instantiation of the *Physical Asset Profile*. In this instantiation, the chassis1 has two slots: slot1 and slot2. The slots are compatible with any type of XYZ:HW:1235Fan packages, as advertised through the CIM_Slot.VendorCompatibilityStrings property. slot1 and package1, which is plugged into it, are compatible because the Delimited Substring matches for the VendorCompatibilityStrings property. slot2 and package2, which is plugged into it, are compatible because an element in the VendorCompatibilityStrings property of the CIM_Slot instance is a Delimited Substring of the element in the VendorCompatibilityStrings property of the CIM_PhysicalPackage instance. chassis1 also has a representation of its fan configuration capacity through capacity1. capacity1 indicates that chassis1 can have a maximum of two fans and should have at least one fan.

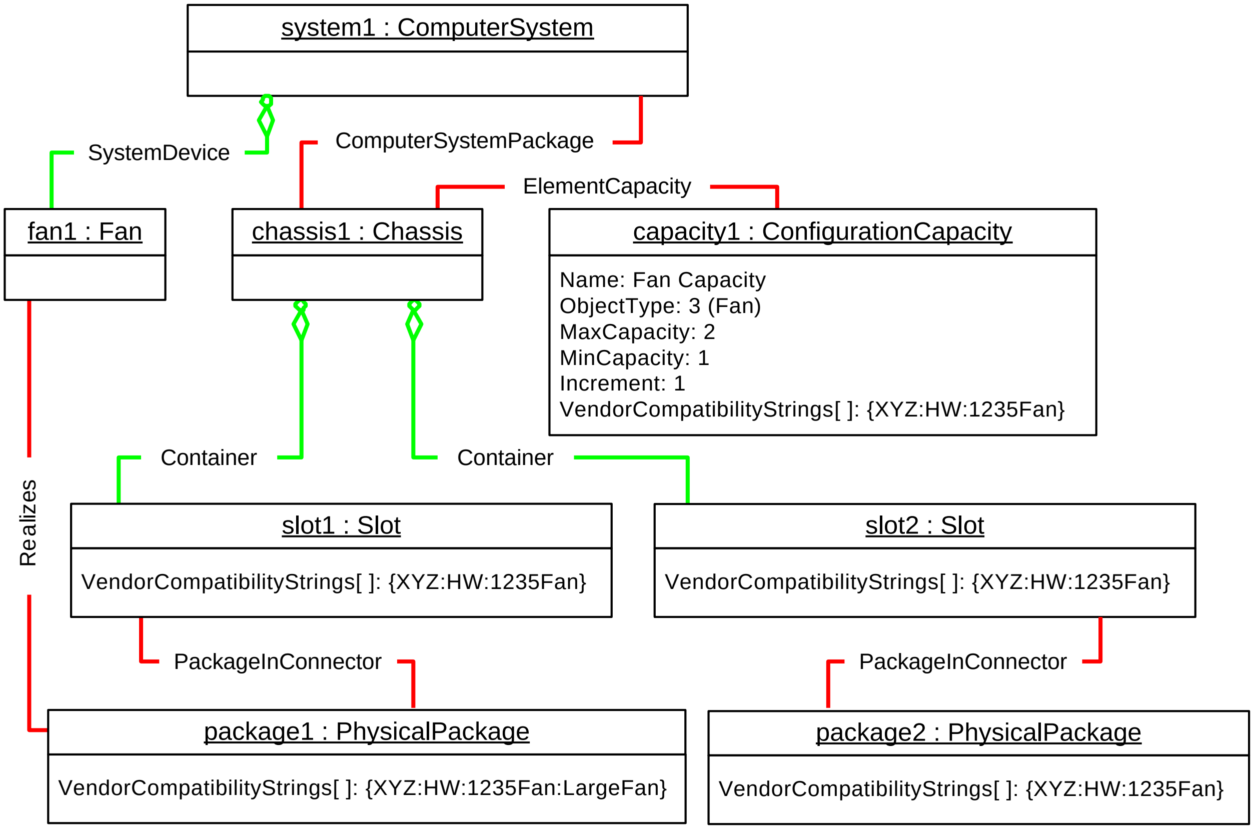

system1 : ComputerSystem
SystemDevice
ComputerSystemPackage
ElementCapacity
fan1 : Fan
chassis1 : Chassis
capacity1 : ConfigurationCapacity
Name: Fan Capacity
ObjectType: 3 (Fan)
MaxCapacity: 2
MinCapacity: 1
Increment: 1
VendorCompatibilityStrings[ ]: {XYZ:HW:1235Fan}
Realizes
Container
Container
slot1 : Slot
VendorCompatibilityStrings[ ]: {XYZ:HW:1235Fan}
slot2 : Slot
VendorCompatibilityStrings[ ]: {XYZ:HW:1235Fan}
PackageInConnector
PackageInConnector
package1 : PhysicalPackage
VendorCompatibilityStrings[ ]: {XYZ:HW:1235Fan:LargeFan}
package2 : PhysicalPackage
VendorCompatibilityStrings[ ]: {XYZ:HW:1235Fan}

Figure 8 – Configuration capacity object diagram

Figure 9 represents another possible instantiation of the *Physical Asset Profile*. In this instantiation, the chassis1 has two cards (card1 and card2) that hold processors. The configuration capacity for card1 is represented by capacity1 because they are associated through the instance of CIM_ElementCapacity. In the same way, card2's configuration capacity is represented by capacity2. Because the VendorCompatibilityStrings property value for capacity1 is equal to the VendorCompatibilityStrings property value for capacity2, the maximum number of compatible processors could be determined by adding the MaxCapacity property value of capacity1 to the MaxCapacity property value of capacity2. In this case, the chassis1 could contain a maximum of four processors.

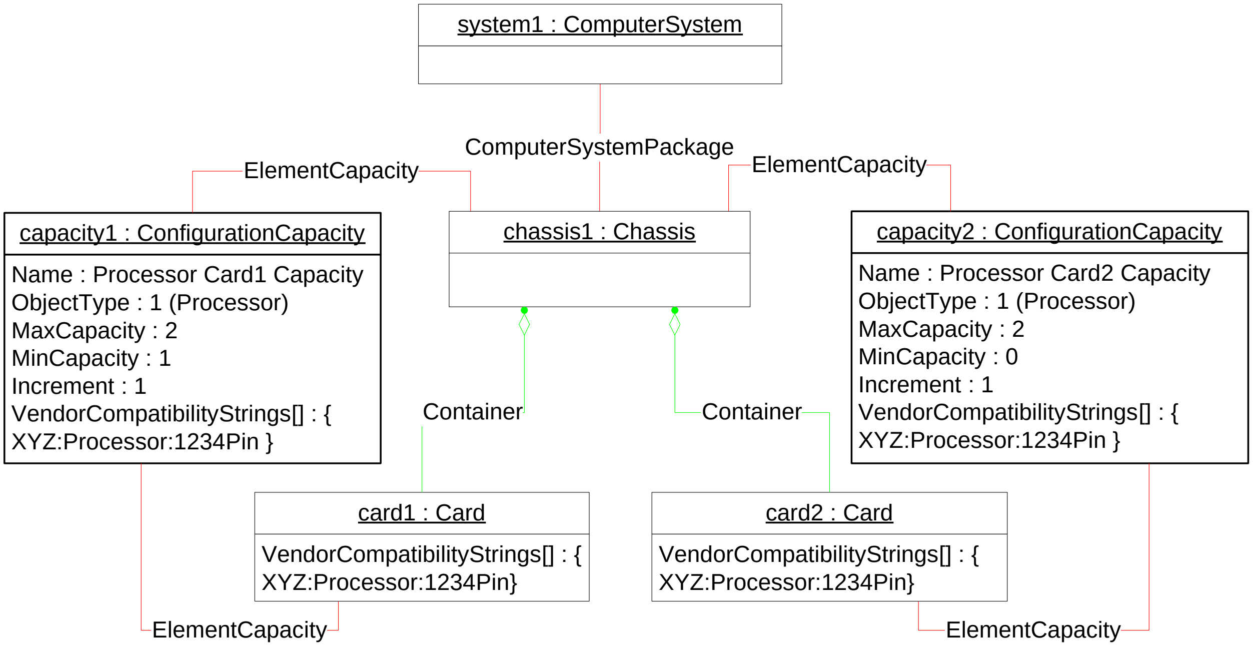


Figure 9 – Additional configuration capacity object diagram

9.8 Representing physical connector

Figure 10 represents another possible instantiation of the *Physical Asset Profile*. In this instance, chassis1 contains a network card, card1. card1 has an RJ45 connector, connector1. connector1 is the physical representation of nic1 network port within system1.

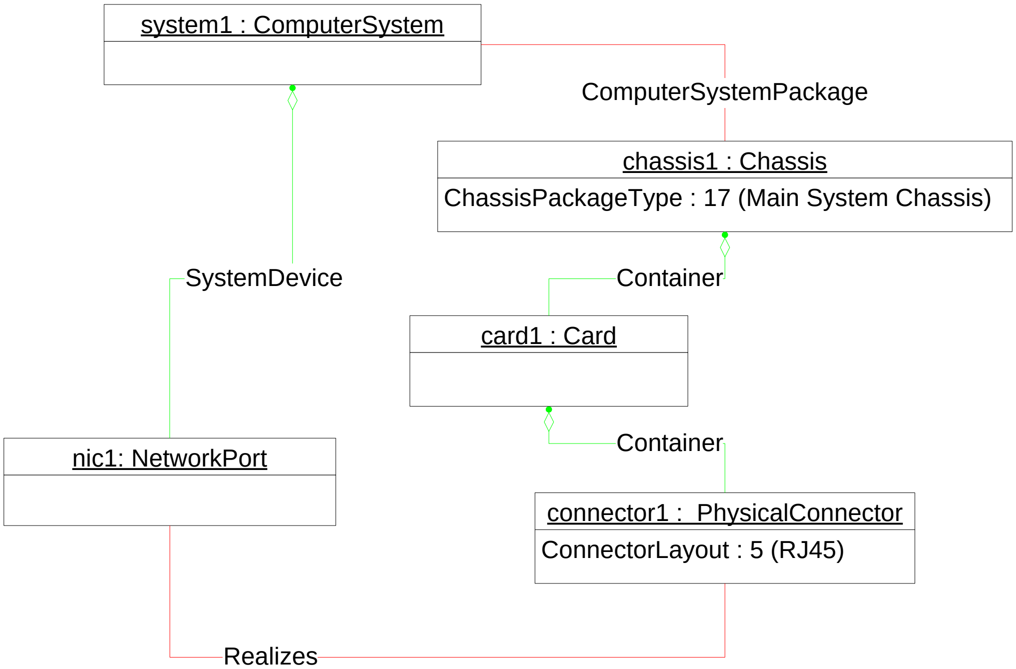


Figure 10 – Network port connector object diagram

9.9 Determining the part number of a failing component

Select the CIM_PhysicalElement subclass instance that is associated through the CIM_Realizes association to the CIM_LogicalDevice component that has a HealthState or OperationalStatus property value indicating that the component is in a failure mode. Get the PartNumber property value for the selected CIM_PhysicalElement subclass instance.

9.10 Obtaining the physical inventory for all devices within a system

Select the CIM_System instance representing the given system. Select all the CIM_LogicalDevice subclass instances that are associated with the CIM_System instance through the CIM_SystemDevice association, and select all the CIM_System instances associated through CIM_SystemComponent associations, and then follow the CIM_SystemDevice association to select all the CIM_LogicalDevice subclass instances. Get all the property values of the CIM_PhysicalElement subclass instances that are associated to the selected CIM_LogicalDevice subclass instances through the CIM_Realizes association and to the selected CIM_System instances through the CIM_SystemPackage association.

9.11 Obtaining the physical inventory for a System Chassis

Get all the property values of the Physical Package instances that are associated through the CIM_SystemPackaging association with the CIM_System instance representing the given system.

9.12 Determining whether the slot is empty

Select all the CIM_ElementInConnector instances that reference the CIM_Slot instance that represents the given slot. If no instances of CIM_ElementInConnector that reference the CIM_Slot instance exist, then the slot is empty; otherwise the slot is occupied by the physical package represented by the instance of CIM_PhysicalPackage referenced by the CIM_ElementInConnector association instance.

9.13 Retrieving the fan capacity for the chassis

For the CIM_Chassis instance that represents the given chassis, select the associated instances of CIM_ConfigurationCapacity through the CIM_ElementCapacity associations. Select CIM_ConfigurationCapacity instances that have the CIM_ConfigurationCapacity.ObjectType property of 3 (Fan).

9.14 Retrieving the maximum capacity of the type of fan package within the chassis

The particular type of fan package is identified through the given string, which is an element of the VendorCompatibilityStrings array property of the Physical Package representing the fan package.

Select all the instances of CIM_ConfigurationCapacity associated with the CIM_Chassis instance through instances of CIM_ElementCapacity where the VendorCompatibilityStrings array property of the instance of CIM_ConfigurationCapacity contains elements equal to the given string. Add all the values for the MaxCapacity property of the selected CIM_ConfigurationCapacity instances.

10 CIM Elements

Table 10 shows the mandatory instances of CIM Elements for this profile. Instances of the following CIM Elements shall be implemented as described in Table 10. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose additional requirements on these elements.

This profile contains definitions for non-abstract parent and child classes. All class definitions are treated as leaf class definitions and the convention used is to replicate the properties in the following tables.

Table 10 – CIM Elements: Physical Asset Profile

Element Name	Requirement	Description
Classes		
CIM_Card	Conditional	See 7.1 and 10.1.
CIM_Chassis	Conditional	See 7.1 and 10.2.
CIM_Chip	Conditional	See 7.1 and 10.3.
CIM_ComputerSystemPackage	Conditional	See 7.1 and 10.4.
CIM_ConfigurationCapacity	Optional	See 7.7 and 10.5.
CIM_ConnectedTo	Optional	See 10.6.
CIM_Container	Optional	See 7.1 and 10.7.
CIM_ElementCapabilities	Conditional	See 10.8.
CIM_ElementCapacity	Conditional	See 7.7 and 10.9.
CIM_ElementInConnector	Optional	See 7.1 and 10.10.
CIM_PhysicalAssetCapabilities	Optional	See 7.4 and 10.11.
CIM_PhysicalComponent	Conditional	See 7.1 and 10.12.
CIM_PhysicalConnector	Conditional	See 7.1 and 10.13.

Element Name	Requirement	Description
CIM_PhysicalFrame	Conditional	See 7.1 and 10.14.
CIM_PhysicalMemory	Conditional	See 7.1 and 10.15.
CIM_PhysicalPackage	Conditional	See 7.1 and 10.16.
CIM_Rack	Conditional	See 7.1 and 10.17.
CIM_Realizes	Conditional	See 7.1 and 10.18.
CIM_RegisteredProfile	Mandatory	See 10.19.
CIM_Slot	Conditional	See 7.1 and 10.20.
CIM_SystemPackaging	Conditional	See 7.1 and 10.21.
Indications		
None defined in this profile		

NOTE Abstract classes are not shown in the tables in the following clauses.

10.1 CIM_Card

CIM_Card represents the processor card and its FRU data. Table 11 contains the requirements for properties of the instance.

Table 11 – Class: CIM_Card

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
HostingBoard	Optional	This property should be implemented.
PackageType	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
VendorCompatibilityStrings	Optional	See 7.5.
Version	Optional	The property shall be the hardware version.
Name	Optional	
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.2 CIM_Chassis

CIM_Chassis represents the chassis and its FRU data. Table 12 contains the requirements for properties of the instance.

Table 12 – Class: CIM_Chassis

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
PackageType	Mandatory	This property shall match 3 (Chassis/Frame).
ChassisPackageType	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
VendorCompatibilityStrings	Optional	See 7.5.
Version	Optional	The property shall be the hardware version.
Name	Optional	
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.3 CIM_Chip

CIM_Chip represents the chip and its FRU data. Table 13 contains the requirements for properties of the instance.

Table 13 – Class: CIM_Chip

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.4 CIM_ComputerSystemPackage

CIM_ComputerSystemPackage associates CIM_ComputerSystem, representing the managed system, with a System Chassis. Table 14 contains the requirements for properties of the instance.

Table 14 – Class: CIM_ComputerSystemPackage

Elements	Requirement	Notes
Antecedent	Mandatory	**Key:** This property shall reference the System Chassis. Cardinality * (indicating zero or many references)
Dependent	Mandatory	**Key:** This property shall reference the CIM_ComputerSystem representing the managed system. Cardinality * (indicating zero or many references)
PlatformGUID	Mandatory	This property shall match "^[0.9A.F]{32}$" or, when unknown, shall match "0" .

10.5 CIM_ConfigurationCapacity

CIM_ConfigurationCapacity advertises the possible configuration of a System Chassis. Table 15 contains the requirements for properties of the instance.

Table 15 – Class: CIM_ConfigurationCapacity

Elements	Requirement	Notes
Name	Mandatory	**Key**
ElementName	Mandatory	None
ObjectType	Mandatory	None
OtherTypeDescription	Conditional	This property shall be implemented when ObjectType matches 0 (Other).
MinimumCapacity	Optional	This property should be implemented.
MaximumCapacity	Mandatory	0 shall mean unknown.
Increment	Mandatory	0 shall mean unknown.
VendorCompatibilityStrings	Optional	See 7.5.

10.6 CIM_ConnectedTo

CIM_ConnectedTo associates the CIM_PhysicalConnector or CIM_Slot instances that represent connectors that are connected together. Table 16 contains the requirements for properties of the instance.

Table 16 – Class: CIM_ConnectedTo

Elements	Requirement	Notes
Antecedent	Mandatory	**Key**: This property shall reference the CIM_PhysicalConnector or CIM_Slot instance. Cardinality * (indicating zero or many references)
Dependent	Mandatory	**Key**: This property shall reference the CIM_PhysicalConnector or CIM_Slot instance. Cardinality * (indicating zero or many references)

10.7 CIM_Container

CIM_Container associates a Physical Package with Physical Elements representing the physical elements that reside within the package. Table 17 contains the requirements for properties of the instance.

Table 17 – Class: CIM_Container

Elements	Requirement	Notes
GroupComponent	Mandatory	**Key:** This property shall reference the Physical Package that represents the container. Cardinality 0.1 (indicating zero or one reference)
PartComponent	Mandatory	**Key:** This property shall reference the Physical Element that is contained within the package. Cardinality * (indicating zero or many references)

10.8 CIM_ElementCapabilities

CIM_ElementCapabilities associates Physical Elements with the CIM_PhysicalAssetCapabilities instances that advertise the physical capabilities. CIM_ElementCapabilities shall be instantiated when an instance of CIM_PhysicalAssetCapabilities exists. Table 18 contains the requirements for properties of the instance.

Table 18 – Class: CIM_ElementCapabilities

Elements	Requirement	Notes
ManagedElement	Mandatory	**Key:** This property shall reference the Physical Element. Cardinality 1.* (indicating one or many references)
Capabilities	Mandatory	**Key:** This property shall reference the CIM_PhysicalAssetCapabilities class. Cardinality 0.1 (indicating zero or one reference)

10.9 CIM_ElementCapacity

CIM_ElementCapacity associates CIM_ConfigurationCapacity instances with a System Chassis. Table 19 contains the requirements for properties of the instance.

Table 19 – Class: CIM_ElementCapacity

Elements	Requirement	Notes
Capacity	Mandatory	**Key**: This property shall reference the CIM_ConfigurationCapacity instance. Cardinality * indicating zero or many references
Element	Mandatory	**Key**: This property shall reference the System Chassis or Physical Package. Cardinality 1.* (indicating one or many references)

10.10 CIM_ElementInConnector

CIM_ElementInConnector associates a CIM_PhysicalConnector or CIM_Slot instance, representing the connector or slot, with Physical Packages (instances of CIM_PhysicalPackage or CIM_PhysicalComponent). Table 20 contains the requirements for properties of the instance.

Table 20 – Class: CIM_ElementInConnector

Elements	Requirement	Notes
Antecedent	Mandatory	**Key:** This property shall reference the instance of CIM_PhysicalConnector or CIM_Slot. Cardinality * (indicating zero or many references)
Dependent	Mandatory	**Key:** This property shall reference the CIM_PhysicalPackage or CIM_PhysicalComponent. Cardinality 0.1 (indicating zero or one reference)

10.11 CIM_PhysicalAssetCapabilities

CIM_PhysicalAssetCapabilities advertises whether the associated instance of a CIM_PhysicalElement subclass contains FRU data. Table 21 contains the requirements for properties of the instance.

Table 21 – Class: CIM_PhysicalAssetCapabilities

Elements	Requirement	Notes
InstanceID	Mandatory	**Key**
ElementName	Mandatory	None
FRUInfoSupported	Mandatory	See 7.4.

10.12 CIM_PhysicalComponent

CIM_PhysicalComponent represents any physical element that cannot be further decomposed, such as ASIC or tape, and its FRU data. Table 22 contains the requirements for properties of the instance.

Table 22 – Class: CIM_PhysicalComponent

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.13 CIM_PhysicalConnector

CIM_PhysicalConnector represents the physical connector. Table 23 contains the requirements for properties of the instance.

Table 23 – Class: CIM_PhysicalConnector

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
ConnectorLayout	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.14 CIM_PhysicalFrame

CIM_PhysicalFrame represents the frame and its FRU data. Table 24 contains the requirements for properties of the instance.

Table 24 – Class: CIM_PhysicalFrame

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
PackageType	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
VendorCompatibilityStrings	Optional	See 7.5.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
Version	Optional	The property shall be the hardware version.
Name	Optional	
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.15 CIM_PhysicalMemory

CIM_PhysicalMemory represents the physical memory and its FRU data. Table 25 contains the requirements for properties of the instance.

Table 25 – Class: CIM_PhysicalMemory

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
FormFactor	Mandatory	None
MemoryType	Mandatory	None
Speed	Mandatory	None
Capacity	Mandatory	None
BankLabel	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.16 CIM_PhysicalPackage

CIM_PhysicalPackage represents the physical package and its FRU data. Table 26 contains the requirements for properties of the instance.

Table 26 – Class: CIM_PhysicalPackage

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
PackageType	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
VendorCompatibilityStrings	Optional	See 7.5.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
Version	Optional	The property shall be the hardware version.
Name	Optional	
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.17 CIM_Rack

CIM_Rack represents the rack and its FRU data. Table 27 contains the requirements for properties of the instance.

Table 27 – Class: CIM_Rack

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
TypeOfRack	Mandatory	None
PackageType	Mandatory	This property shall match 2 (Rack).
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
VendorCompatibilityStrings	Optional	See 7.5.
CanBeFRUed	Optional	This property should be implemented when the Physical Element can be replaced in the field.
Version	Optional	The property shall be the hardware version.
Name	Optional	
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.18 CIM_Realizes

CIM_Realizes associates an instance of a CIM_LogicalDevice subclass, representing the logical device, with a Physical Element. Table 28 contains the requirements for properties of the instance.

Table 28 – Class: CIM_Realizes

Elements	Requirement	Notes
Antecedent	Mandatory	**Key:** shall reference the Physical Element. Cardinality 1.* indicating one or many references
Dependent	Mandatory	**Key:** shall reference the instance of subclass of CIM_LogicalDevice Cardinality * indicating zero or many references

10.19 CIM_RegisteredProfile

The CIM_RegisteredProfile class is defined by the *Profile Registration Profile*. Table 29 contains the requirements for properties of the class.

The requirements listed in Table 29 are in addition to those mandated by the *Profile Registration Profile*.

Table 29 – Class: CIM_RegisteredProfile

Elements	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of “Physical Asset”.
RegisteredVersion	Mandatory	This property shall have a value of “1.0.3”.
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.

10.20 CIM_Slot

CIM_Slot represents the slot and its FRU data. Table 30 contains the requirements for properties of the instance.

Table 30 – Class: CIM_Slot

Elements	Requirement	Notes
Tag	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
Number	Mandatory	None
ConnectorLayout	Mandatory	None
Manufacturer	Conditional	See 7.4.
Model	Conditional	See 7.4.
SerialNumber	Conditional	See 7.4.
PartNumber	Conditional	See 7.4.
SKU	Conditional	See 7.4.
VendorCompatibilityStrings	Optional	See 7.5.
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern “.*”).

10.21 CIM_SystemPackaging

CIM_SystemPackaging associates CIM_System, which represents the managed system, with a System Chassis. Table 31 contains the requirements for properties of the instance.

Table 31 – Class: CIM_SystemPackaging

Elements	Requirement	Notes
Antecedent	Mandatory	**Key:** This property shall reference the System Chassis. Cardinality * (indicating zero or many references)
Dependent	Mandatory	**Key:** This property shall reference the CIM_System representing the managed system. Cardinality * (indicating zero or many references)

ANNEX A
(informative)

Change log

Version	Date	Description
1.0.0	2007-12-11	
1.0.1	2008-06-09	Incorporated errata submitted for the Final Standard.
1.0.2	2009-04-06	DMTF Standard Release Incorporated errata on CIM_PhysicalMemory.Speed property values for unknown or variable speeds.
1.0.3	2016-05-17	Incorporated errata for clarifying white space related requirements for FRU properties.