1

5 # Physical Computer System View Profile

34   CONTENTS

71

72 **Figures**

78

Tables

106   Foreword

107   The *Physical Computer System View Profile* (DSP1108) was prepared by the Server Desktop Mobile
108   Platforms Working Group of the DMTF.

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

Acknowledgments

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

113   Editors:

114   • Hemal Shah – Broadcom Limited

115   • Steve Lee – Microsoft Corporation

116   Contributors:

117   • Jeff Hilland – Hewlett-Packard Company

118   • John Leung – Intel Corporation

119   • Hemal Shah – Broadcom Corporation

120   • Satheesh Thomas – AMI

121   • Perry Vincent – Intel Corporation

122   • Divyanand Malavalli - Advanced Micro Devices

123   Introduction

124   The information in this specification should be sufficient for a provider or consumer of this data to identify
125   unambiguously the classes, properties, methods, and values that shall be instantiated and manipulated to
126   represent and manage a physical computer system and its associated management information.

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

129   **Document conventions**

130   **Typographical conventions**

131   The following typographical conventions are used in this document:

132   • Document titles are marked in *italics*.

133   • Important terms that are used for the first time are marked in *italics*.

134   • Terms include a link to the term definition in the "Terms and definitions" clause, enabling easy
135   navigation to the term definition.

136   • ABNF rules are in `monospaced font`.

137   **ABNF usage conventions**

138   Format definitions in this document are specified using ABNF (see RFC5234), with the following
139   deviations:

140   • Literal strings are to be interpreted as case-sensitive Unicode characters, as opposed to the
141   definition in RFC5234 that interprets literal strings as case-insensitive US-ASCII characters.

142 # Physical Computer System View Profile

143 ## 1   Scope

144 The *Physical Computer System View Profile* describes a view of the management capability of
145 referencing profiles by adding the capability to represent a physical computer system view of a managed
146 computer system. This profile includes a specification of the physical computer system view, extrinsic
147 methods for management operations, and its associated relationships to referencing profiles. This profile
148 is not intended to provide all details of referenced profiles.

149 ## 2   Normative references

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

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

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

158 DMTF DSP0223, *Generic Operations 1.0*,
159 http://www.dmtf.org/standards/published_documents/DSP0223_1.0.pdf

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

162 DMTF DSP1009, *Sensors Profile 1.0*,
163 http://dmtf.org/sites/default/files/standards/documents/DSP1009_1.0.pdf

164 DMTF DSP1009, *Sensors Profile 1.1*,
165 http://dmtf.org/sites/default/files/standards/documents/DSP1009_1.1.pdf

166 DMTF DSP1010, *Record Log Profile 1.0*,
167 http://www.dmtf.org/sites/default/files/standards/documents/DSP1010_1.0.pdf

168 DMTF DSP1010, *Record Log Profile 2.0*,
169 http://www.dmtf.org/sites/default/files/standards/documents/DSP1010_2.0.pdf

170 DMTF DSP1011, *Physical Asset Profile 1.0,*
171 http://www.dmtf.org/sites/default/files/standards/documents/DSP1011_1.0.pdf

172 DMTF DSP1012, *Boot Control Profile 1.0*,
173 http://dmtf.org/sites/default/files/standards/documents/DSP1012_1.0.pdf

174 DMTF DSP1022, *CPU Profile 1.0*,
175 http://dmtf.org/sites/default/files/standards/documents/DSP1022_1.0.pdf

176 DMTF DSP1023, *Software Inventory Profile 1.0*,
177 http://dmtf.org/sites/default/files/standards/documents/DSP1023_1.0.pdf

178 DMTF DSP1025, *Software Update Profile 1.0*,
179 http://dmtf.org/sites/default/files/standards/documents/DSP1025_1.0.pdf

180 DMTF DSP1026, *System Memory Profile 1.0*,
181 http://dmtf.org/sites/default/files/standards/documents/DSP1026_1.0.pdf

182 DMTF DSP1029, *OS Status Profile 1.0,*
183 http://dmtf.org/sites/default/files/standards/documents/DSP1029_1.0.pdf

184 DMTF DSP1029, *OS Status Profile 1.1,*
185 http://dmtf.org/sites/default/files/standards/documents/DSP1029_1.1.pdf

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

188 DMTF DSP1033, *Profile Registration Profile 1.1,*
189 http://www.dmtf.org/standards/published_documents/DSP1033_1.1.pdf

190 DMTF DSP1052, *Computer System Profile 1.0,*
191 http://www.dmtf.org/sites/default/files/standards/documents/DSP1052_1.0.pdf

192 DMTF DSP1085, *Power Utilization Management Profile 1.0*,
193 http://dmtf.org/sites/default/files/standards/documents/DSP1085_1.0.pdf

194 IETF RFC5234, *ABNF: Augmented BNF for Syntax Specifications, January 2008*,
195 http://tools.ietf.org/html/rfc5234

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

198

199 3   Terms and definitions

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

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

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

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

213   The terms defined in DSP0004, DSP0223, DSP1001, and DSP1033 apply to this document. The
214   following additional terms are used in this document.

215   **3.1**
216   **conditional**
217   indicates requirements to be followed strictly to conform to the document when the specified conditions
218   are met

219   **3.2**
220   **mandatory**
221   indicates requirements to be followed strictly to conform to the document and from which no deviation is
222   permitted

223   **3.3**
224   **optional**
225   indicates a course of action permissible within the limits of the document

226   **3.4**
227   **referencing profile**
228   indicates a profile that owns the definition of this class and can include a reference to this profile in its
229   "Referenced Profiles" table

230   **3.5**
231   **unspecified**
232   indicates that this profile does not define any constraints for the referenced CIM element or operation

233   **4   Symbols and abbreviated terms**

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

236   **4.1**
237   **BIOS**
238   Basic Input Output System

239   **4.2**
240   **EFI**
241   Extensible Firmware Interface

242   **4.3**
243   **FRU**
244   Field Replaceable Unit

245   **4.4**
246   **SKU**
247   Stock Keeping Unit

248

249 5 Synopsis

250 **Profile name:** Physical Computer System View

251 **Version:** 1.0.3

252 **Organization:** DMTF

253 **CIM schema version:** 2.36

254 **Central class:** CIM_PhysicalComputerSystemView

255 **Scoping class:** CIM_ComputerSystem

256 The *Physical Computer System View Profile* extends the management capability of referencing profiles
257 by adding the capability to represent a physical computer system view of a managed computer system.
258 This profile includes a specification of the physical computer system view, extrinsic methods for
259 management operations, and its associated relationships to referencing profiles.

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

261 **Table 1 – Referenced profiles**

Profile Name	Organization	Version	Requirement	Description
Profile Registration	DMTF	1.0	Mandatory	None
Profile Registration	DMTF	1.1	Optional	None
Computer System	DMTF	1.0	Optional	None
Power Utilization Management	DMTF	1.0	Optional	None
Sensors	DMTF	1.0	Optional	None
Sensors	DMTF	1.1	Optional	None
Record Log	DMTF	1.0	Optional	None
Boot Control	DMTF	1.0	Optional	None
Software Inventory	DMTF	1.0	Optional	None
System Memory	DMTF	1.0	Optional	None
Physical Asset	DMTF	1.0	Optional	None
OS Status	DMTF	1.0	Optional	None
OS Status	DMTF	1.1	Optional	None
CPU	DMTF	1.0	Optional	None
Software Update	DMTF	1.0	Optional	None

262 # 6   Description

263 The *Physical Computer System View Profile* describes a physical computer system and associated
264 management information in a managed computer system.

265 Figure 1 represents the class schema for the *Physical Computer System View Profile.* For simplicity, the
266 CIM_ prefix has been removed from the names of the classes.
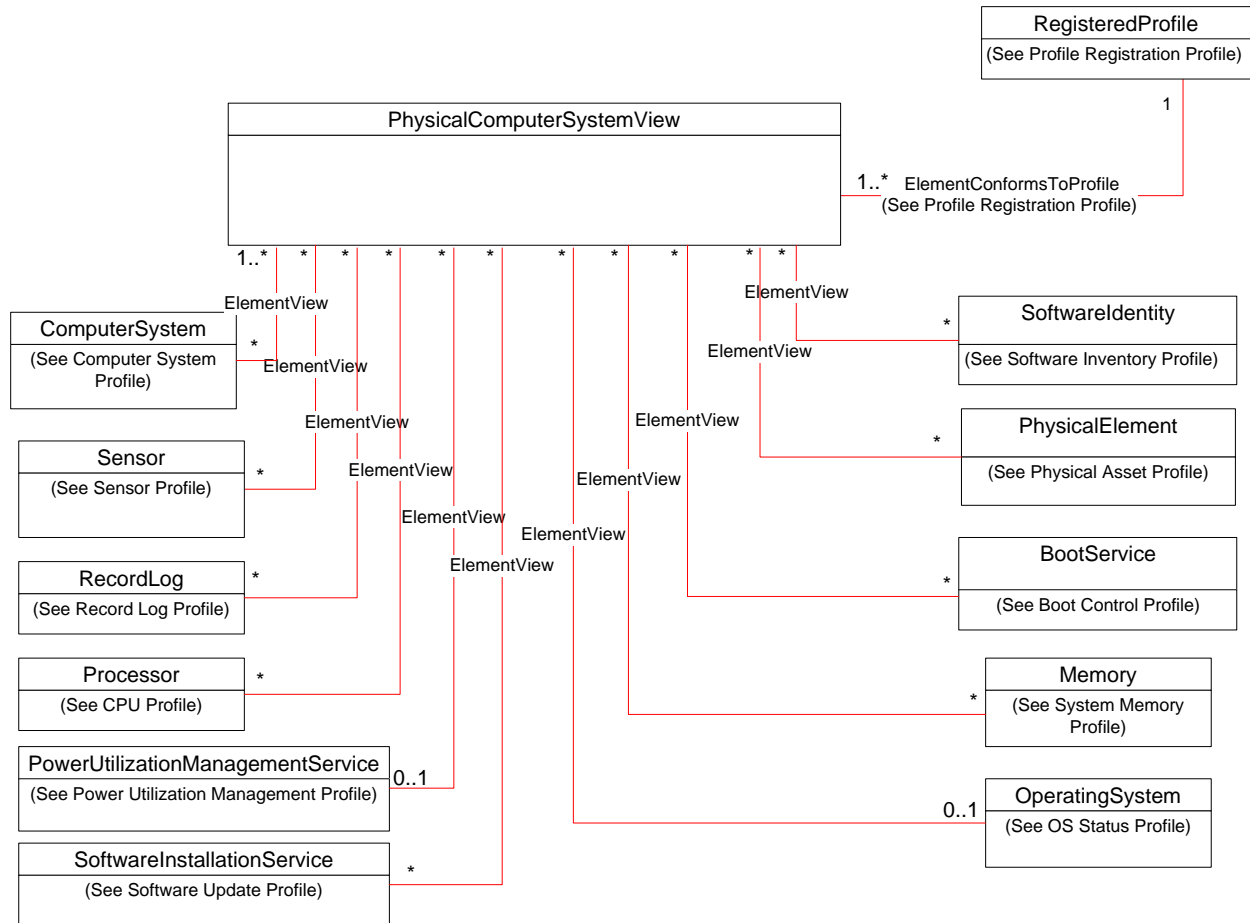
267

268 **Figure 1 – Physical Computer System View Profile: Class diagram**

269 # 7   Implementation

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

272 ## 7.1   Representing a physical computer system view

273 A view of the managed computer system shall be represented by an instance of
274 CIM_PhysicalComputerSystemView. The properties of the instance of
275 CIM_PhysicalComputerSystemView shall reflect the current state and configuration of a managed
276 computer system conforming to the referenced profiles.

277   When the ImplementedFeatures property of the CIM_RegisteredProfile instance contains the value listed
278   in Table 2 below, the corresponding CIM_PhysicalComputerSystemView property shall conform to
279   requirements for the corresponding property listed in the table as specified by the referenced profile.

7.1.1   Representing information from multiple instances of the same class

281   A subset of properties for CIM_PhysicalComputerSystemView may represent information from multiple
282   instances of same class. Indexed Arrays are used in CIM_PhysicalComputerSystemView for such
283   properties. The value in the array correspondence column in Table 2 identifies the class whose multiple
284   instances are represented by the corresponding array property in CIM_PhysicalComputerSystemView.

7.1.1.1   Properties of each instance

286   The same index in properties with array correspondence with the same class shall represent a view of a
287   single instance of that class.

7.1.1.2   Property correspondence

289   The elements of properties with array correspondence with a class shall correspond to the respective
290   properties of the corresponding instance of that class as in Table 2.

7.1.1.3   Matching property values to normalized instances

292   For all properties with array correspondence with a class as in Table 2, the value at each index shall
293   match the value of the corresponding property of the corresponding instance of that class.

294   **Table 2 – CIM_PhysicalComputerSystemView property model correspondence**

Implemented Feature Value	CIM_Physical ComputerSystem View Property	Origin Class/Property or Model Correspondence	Referenced Profile
DMTF:Physical AssetView	FRUInfoSupported	CIM_PhysicalAssetCapabilities.FRUInfoSupported for CIM_Chassis	DSP1011 1.0 Clause 10.2
	Tag	CIM_Chassis.Tag	DSP1011 1.0 Clause 10.2
	Manufacturer	CIM_Chassis.Manufacturer	DSP1011 1.0 Clause 10.2
	Model	CIM_Chassis.Model	DSP1011 1.0 Clause 10.2
	SKU	CIM_Chassis.SKU	DSP1011 1.0 Clause 10.2
	SerialNumber	CIM_Chassis.SerialNumber	DSP1011 1.0 Clause 10.2
	Version	CIM_Chassis.Version	DSP1011 1.0 Clause 10.2
	PartNumber	CIM_Chassis.PartNumber	DSP1011 1.0 Clause 10.2
DMTF:CPUView	NumberOfProcessors	Number of CIM_Processor instances associated to associated CIM_ComputerSystem	DSP1022 1.0 Clause 10.11
	NumberOfProcessor Cores	CIM_ProcessorCapabilities.NumberOfProcessorCores	DSP1022 1.0 Clause 10.12
	NumberOfProcessor Threads	CIM_ProcessorCapabilities.NumberOfHardwareThreads	DSP1022 1.0 Clause 10.12
	ProcessorFamily	CIM_Processor.Family	DSP1022 1.0 Clause 10.11
	ProcessorCurrent ClockSpeed	CIM_Processor.CurrentClockSpeed	DSP1022 1.0 Clause 10.11
	ProcessorMaxClock Speed	CIM_Processor.MaxClockSpeed	DSP1022 1.0 Clause 10.11
DMTF:System MemoryView	MemoryBlockSize	CIM_Memory.BlockSize	DSP1026 1.0 Clause 10.3
	MemoryNumberOf Blocks	CIM_Memory.NumberOfBlocks	DSP1026 1.0 Clause 10.3
	MemoryConsumable Blocks	CIM_Memory.ConsumableBlocks	DSP1026 1.0 Clause 10.3
DMTF:Software InventoryView	CurrentBIOSMajor Version	CIM_SoftwareIdentity.MajorVersion representing the BIOS/EFI	DSP1023 1.0 Clause 10.1
	CurrentBIOSMinor Version	CIM_SoftwareIdentity.MinorVersion representing the BIOS/EFI	DSP1023 1.0 Clause 10.1
	CurrentBIOSRevision Number	CIM_SoftwareIdentity.RevisionNumber representing the BIOS/EFI	DSP1023 1.0 Clause 10.1

Implemented Feature Value	CIM_Physical ComputerSystem View Property	Origin Class/Property or Model Correspondence	Referenced Profile
	CurrentBIOSBuild Number	CIM_SoftwareIdentity.BuildNumber representing the BIOS/EFI	DSP1023 1.0 Clause 10.1
	CurrentBIOSVersion String	CIM_SoftwareIdentity.VersionString representing the BIOS/EFI	DSP1023 1.0 Clause 10.1
	CurrentManagement Firmware MajorVersion	CIM_SoftwareIdentity.MajorVersion representing the management firmware	DSP1023 1.0 Clause 10.1
	CurrentManagement Firmware MinorVersion	CIM_SoftwareIdentity.MinorVersion representing the management firmware	DSP1023 1.0 Clause 10.1
	CurrentManagement Firmware RevisionNumber	CIM_SoftwareIdentity.Revision Number representing the management firmware	DSP1023 1.0 Clause 10.1
	CurrentManagement Firmware BuildNumber	CIM_SoftwareIdentity.BuildNumber representing the management firmware	DSP1023 1.0 Clause 10.1
	CurrentManagementFi rmwareElementName	CIM_SoftwareIdentity.Element Name representing the management firmware	DSP1023 1.0 Clause 10.1
	CurrentManagement Firmware VersionString	CIM_SoftwareIdentity.VersionString representing the management firmware	DSP1023 1.0 Clause 10.1
DMTF:OSView	OSType	CIM_OperatingSystem.OSType	DSP1029 1.0 Clause 10.1
	OSEnabledState	CIM_OperatingSystem.Enabled State	DSP1029 1.0 Clause 10.1
	OSVersion	CIM_OperatingSystem.Version	DSP1029 1.1 Clause 10.1
DMTF:Computer SystemView	OtherIdentifyingInfo	CIM_ComputerSystem.Other IdentifyingInfo	DSP1052 1.0 Clause 10.1
	IdentifyingDescriptions	CIM_ComputerSystem.Identifying Descriptions	DSP1052 1.0 Clause 10.1
	Dedicated	CIM_ComputerSystem.Dedicated	DSP1108 1.0 Clause 7.1.2.10.1
	OtherDedicated Descriptions	CIM_ComputerSystem.Other DedicatedDescriptions	DSP1108 1.0 Clause 7.1.2.10.2
	EnabledState	CIM_ComputerSystem.Enabled State	DSP1052 1.0 Clause 10.1
	RequestedState	CIM_ComputerSystem.Requested State	DSP1052 1.0 Clause 10.1

Implemented Feature Value	CIM_Physical ComputerSystem View Property	Origin Class/Property or Model Correspondence	Referenced Profile
	OperationalStatus	CIM_ComputerSystem.Operational Status	DSP1052 1.0 Clause 10.1
	HealthState	CIM_ComputerSystem.HealthState	DSP1052 1.0 Clause 10.1
DMTF:Power Utilization ManagementView	PowerUtilizationModes Supported	CIM_PowerUtilizationManagement Capabilities.PowerUtilizationModes Supported	DSP1085 1.0 Clause 10.2
	PowerUtilizationMode	CIM_PowerUtilizationManagement Service.PowerUtilizationMode	DSP1085 1.0 Clause 10.1
	PowerAllocationLimit	CIM_PowerAllocationSettingData. Limit	DSP1085 1.0 Clause 10.6
DMTF:Numeric SensorView	NumericSensor ElementName	CIM_NumericSensor.ElementName	DSP1009 1.0 Clause 10.2
	NumericSensor EnabledState	CIM_NumericSensor.EnabledState	DSP1009 1.0 Clause 10.2
	NumericSensor Context	CIM_NumericSensor.Sensor Context	DSP1009 1.1 Clause 10.2
	NumericSensorHealth State	CIM_NumericSensor.HealthState	DSP1009 1.0 Clause 10.2
	NumericSensor CurrentState	CIM_NumericSensor.CurrentState	DSP1009 1.0 Clause 10.2
	NumericSensor PrimaryStatus	CIM_NumericSensor.PrimaryStatus	DSP1009 1.0 Clause 10.2
	NumericSensorBase Units	CIM_NumericSensor.BaseUnits	DSP1009 1.0 Clause 10.2
	NumericSensorUnit Modifier	CIM_NumericSensor.UnitModifier	DSP1009 1.0 Clause 10.2
	NumericSensorRate Units	CIM_NumericSensor.RateUnits	DSP1009 1.0 Clause 10.2
	NumericSensor CurrentReading	CIM_NumericSensor.Current Reading	DSP1009 1.0 Clause 10.2
	NumericSensorSensor Type	CIM_NumericSensor.SensorType	DSP1009 1.0 Clause 10.2
	NumericSensorOther SensorType Description	CIM_NumericSensor.OtherSensor TypeDescription	DSP1009 1.0 Clause 10.2
	NumericSensorUpper ThresholdNonCritical	CIM_NumericSensor.Upper ThresholdNonCritical	DSP1009 1.0 Clause 10.2
	NumericSensorUpper ThresholdFatal	CIM_NumericSensor.Upper ThresholdFatal	DSP1009 1.0 Clause 10.2
	NumericSensorUpper ThresholdCritical	CIM_NumericSensor.Upper ThresholdCritical	DSP1009 1.0 Clause 10.2

Implemented Feature Value	CIM_Physical ComputerSystem View Property	Origin Class/Property or Model Correspondence	Referenced Profile
	NumericSensorLower ThresholdNonCritical	CIM_NumericSensor.Lower ThresholdNonCritical	DSP1009 1.0 Clause 10.2
	NumericSensorLower ThresholdFatal	CIM_NumericSensor.Lower ThresholdFatal	DSP1009 1.0 Clause 10.2
	NumericSensorLower ThresholdCritical	CIM_NumericSensor.Lower ThresholdCritical	DSP1009 1.0 Clause 10.2
DMTF:Record LogView	LogInstanceID	CIM_RecordLog.InstanceID	DSP1010 1.0 Clause 10.5
	LogMaxNumberOf Records	CIM_RecordLog.MaxNumberOf Records	DSP1010 1.0 Clause 10.5
	LogCurrentNumberOf Records	CIM_RecordLog.CurrentNumberOf Records	DSP1010 1.0 Clause 10.5
	LogOverWritePolicy	CIM_RecordLog.OverwritePolicy	DSP1010 1.0 Clause 10.5
	LogState	CIM_RecordLog.LogState	DSP1010 1.0 Clause 10.5
DMTF:Boot ControlView	StructuredBootString	CIM_BootSourceSetting.Structured BootString	DSP1012 1.0 Clause 10.6
	OneTimeBootSource	n/a	DSP1108 1.0 Clause 7.1.2.9.2

7.1.2 Additional requirements

296 This subclause details additional requirements for some properties of
297 CIM_PhysicalComputerSystemView.

7.1.2.1 CIM_PhysicalComputerSystemView.InstanceID

299 The InstanceID is the property that shall be used to opaquely and uniquely identify an instance of this
300 class within the scope of the instantiating Namespace. This property shall not correspond to the
301 InstanceID property of CIM_ComputerSystem.

7.1.2.2 Representing system processor information

303 When implemented according to Table 2, the intent of this set of properties is to model the central
304 processing unit.

305 The NumberOfProcessors property represents the number of homogenous processors on this physical
306 computer system. Other types of processors (including GPUs) shall not be represented in the
307 NumberOfProcessors property of CIM_PhysicalComputerSystemView.

308 The NumberOfProcessorThreads property shall correspond to the NumberOfHardwareThreads property
309 in the CIM_ProcessorCapabilities class defined in DSP1022.

310 The properties of the central processing unit of the physical computer system shall be represented as
311 properties defined in Table 3 from DSP1022.
312

313 When one or more instances of CIM_Processor are instantiated and represented in the view class, each
314 CIM_Processor instance should be associated with the CIM_PhysicalComputerSystemView instance
315 through an instance of CIM_ElementView where the Antecedent property is a reference to the
316 corresponding CIM_Processor instance and the Dependent property is a reference to the
317 CIM_PhysicalComputerSystemView instance.

318   **Table 3 – Property origins for processor**

CIM_PhysicalComputerSystemView property name	CIM_Processor property name (origin)
ProcessorFamily	Family
ProcessorMaxClockSpeed	MaxClockSpeed

319 **7.1.2.3   Representing system memory information**

320 When implemented according to Table 2, the properties of the memory of the physical computer system
321 shall be represented as properties defined in Table 4 from DSP1026. When the instance of CIM_Memory
322 representing total system memory is instantiated and represented in the view class, the CIM_Memory
323 instance should be associated with the CIM_PhysicalComputerSystemView instance through an instance
324 of CIM_ElementView where the Antecedent property is a reference to the CIM_Memory instance and the
325 Dependent property is a reference to the CIM_PhysicalComputerSystemView instance.

326   **Table 4 – Property origins for memory**

CIM_PhysicalComputerSystemView property name	CIM_Memory property name (origin)
MemoryBlockSize	BlockSize
MemoryNumberOfBlocks	NumberOfBlocks
MemoryConsumableBlocks	ConsumableBlocks

327 **7.1.2.4   Representing system software inventory**

328 When implemented according to Table 2, the properties of the current BIOS or EFI firmware of the
329 physical computer system shall be represented as properties defined in Table 5. The current BIOS or EFI
330 firmware property values shall correspond to an instance of CIM_SoftwareIdentity where the
331 Classifications property contains a value of 10 (Firmware) or 11 (BIOS/FCode). If instantiated, this
332 corresponding instance of CIM_SoftwareIdentity shall be associated with the underlying instance of
333 CIM_ComputerSystem by an instance of CIM_ElementSoftwareIdentity where the
334 ElementSoftwareStatus property has a value of 2 (Current).

335 When an instance of CIM_SoftwareIdentity representing the current BIOS or EFI firmware is instantiated
336 and represented in the view class, the CIM_SoftwareIdentity instance should be associated with the
337 CIM_PhysicalComputerSystemView instance through an instance of CIM_ElementView where the
338 Antecedent is a reference to the CIM_SoftwareIdentity instance and the Dependent property is a
339 reference to the CIM_PhysicalComputerSystemView instance.

340 **Table 5 – Property origins for current BIOS or EFI firmware**

CIM_PhysicalComputerSystemView property name	CIM_SoftwareIdentity property name (origin)
CurrentBIOSMajorVersion	MajorVersion
CurrentBIOSMinorVersion	MinorVersion
CurrentBIOSRevisionNumber	RevisionNumber
CurrentBIOSBuildNumber	BuildNumber
CurrentBIOSVersionString	VersionString

341 The properties of the current management firmware of the physical computer system shall be represented
342 as properties defined in Table 6 from DSP1023. The current management firmware property values shall
343 correspond to the instance of CIM_SoftwareIdentity referenced by the instance of
344 CIM_ElementSoftwareIdentity where the ElementSoftwareStatus property has a value of 2 (Current).

345 When an instance of CIM_SoftwareIdentity representing the current management firmware is instantiated
346 and represented in the view class, the CIM_SoftwareIdentity instance should be associated with the
347 CIM_PhysicalComputerSystemView instance through an instance of CIM_ElementView where the
348 Antecedent is a reference to the CIM_SoftwareIdentity instance and the Dependent property is a
349 reference to the CIM_PhysicalComputerSystemView instance.

350 **Table 6 – Property origins for current management firmware**

CIM_PhysicalComputerSystemView property name	CIM_SoftwareIdentity property name (origin)
CurrentManagementFirmwareMajorVersion	MajorVersion
CurrentManagementFirmwareMinorVersion	MinorVersion
CurrentManagementFirmwareRevisionNumber	RevisionNumber
CurrentManagementFirmwareBuildNumber	BuildNumber
CurrentManagementFirmwareVersionString	VersionString
CurrentManagementFirmwareElementName	ElementName

351 **7.1.2.5 Representing operating system information**

352 This subclause describes the requirements for representing the running operating system for the
353 CIM_PhysicalComputerSystemView class.

354 When implemented according to Table 2, the properties of the operating system of the physical computer
355 system shall be represented as properties as defined in Table 7 from DSP1029. When an instance of
356 CIM_OperatingSystem is instantiated and represented in the view class, the CIM_OperatingSystem
357 instance should be associated with the CIM_PhysicalComputerSystemView instance through an instance
358 of CIM_ElementView where the Antecedent property is a reference to the CIM_OperatingSystem
359 instance and the Dependent property is a reference to the CIM_PhysicalComputerSystemView instance.

360   **Table 7 – Property origins for the operating system**

CIM_PhysicalComputerSystemView property name	CIM_OperatingSystem property name (origin)
OSEnabledState	EnabledState
OSVersion	Version
OSType	OSType

361 **7.1.2.6   Representing power utilization information**

362 When implemented according to Table 2, the PowerAllocationLimit property shall correspond to the Limit
363 property of the CIM_PowerAllocationSettingData class defined in DSP1085 with the added
364 "PowerAllocation" prefix to logically group properties related to power utilization and avoid naming
365 collision.

366 The power allocation limit of the physical computer system shall be represented as the property as
367 defined in Table 8 from DSP1085. When an instance of CIM_PowerAllocationSettingData is instantiated
368 and represented in the view class, the CIM_PowerUtilizationManagementService instance should be
369 associated with the CIM_PhysicalComputerSystemView instance through an instance of
370 CIM_ElementView where the Antecedent property is a reference to the
371 CIM_PowerUtilizationManagementService instance and the Dependent property is a reference to the
372 CIM_PhysicalComputerSystemView instance.

373   **Table 8 – Property origins for the power allocation limit**

CIM_PhysicalComputerSystemView property name	CIM_PowerAllocationSettingData property name (origin)
PowerAllocationLimit	Limit

374 **7.1.2.7   Representing system numeric sensors**

375 When implemented according to Table 2, the properties of a numeric sensor of the physical computer
376 system shall be represented as elements of a group of indexed array properties as defined in Table 9
377 from DSP1009. For these properties, the array elements with same index shall present a view of the
378 same numeric sensor. When an instance of CIM_NumericSensor is instantiated and represented in the
379 view class, the CIM_NumericSensor instance should be associated with the
380 CIM_PhysicalComputerSystemView instance through an instance of CIM_ElementView where the
381 Antecedent property is a reference to the CIM_NumericSensor instance and the Dependent property is a
382 reference to the CIM_PhysicalComputerSystemView instance.

383 **Table 9 – Property origins for numeric sensors**

CIM_PhysicalComputerSystemView property name	CIM_NumericSensor property name (origin)
NumericSensorElementName[]	ElementName
NumericSensorEnabledState[]	EnabledState
NumericSensorHealthState[]	HealthState
NumericSensorCurrentState[]	CurrentState
NumericSensorPrimaryStatus[]	PrimaryStatus
NumericSensorBaseUnits[]	BaseUnits
NumericSensorUnitModifier[]	UnitModifier
NumericSensorRateUnits[]	RateUnits
NumericSensorCurrentReading[]	CurrentReading
NumericSensorSensorType[]	SensorType
NumericSensorOtherSensorTypeDescription[]	OtherSensorTypeDescription
NumericSensorUpperThresholdNonCritical[]	UpperThresholdNonCritical
NumericSensorUpperThresholdFatal[]	UpperThresholdFatal
NumericSensorUpperThresholdCritical[]	UpperThresholdCritical
NumericSensorLowerThresholdNonCritical[]	LowerThresholdNonCritical
NumericSensorLowerThresholdFatal[]	LowerThresholdFatal
NumericSensorLowerThresholdCritical[]	LowerThresholdCritical
NumericSensorContext[]	SensorContext

384 7.1.2.8 Representing system record logs

385 When implemented according to Table 2, the properties of a record log of the physical computer system
386 shall be represented as elements of a group of indexed array properties as defined in Table 10 from
387 DSP1010. For these properties, the array elements with same index shall present a view of the same
388 record log.

389 Instances of CIM_RecordLog that contain information about the underlying computer system should be
390 represented in the view class.

391 When instantiated, the CIM_RecordLog instance shall be associated with the
392 CIM_PhysicalComputerSystemView instance through an instance of CIM_ElementView where the
393 Antecedent property is a reference to the CIM_RecordLog instance and the Dependent property is a
394 reference to the CIM_PhysicalComputerSystemView instance.

395

Table 10 – Property origins for record logs

CIM_PhysicalComputerSystemView property name	CIM_RecordLog property name (origin)
LogInstanceID[]	InstanceID
LogMaxNumberOfRecords[]	MaxNumberOfRecords
LogCurrentNumberOfRecords[]	CurrentNumberOfRecords
LogOverWritePolicy[]	OverwritePolicy
LogState[]	LogState

396 **7.1.2.9 Representing system boot configuration**

397 When implemented according to Table 2, the enabled boot sources of the boot configuration of the
398 physical computer system shall be represented as elements of an ordered array property as defined in
399 Table 11 from DSP1012.

400 When an instance of CIM_BootSourceSetting is instantiated and represented in the view class, the
401 CIM_BootService instance should be associated with the CIM_PhysicalComputerSystemView instance
402 through an instance of CIM_ElementView where the Antecedent property is a reference to the
403 CIM_BootService instance and the Dependent property is a reference to the
404 CIM_PhysicalComputerSystemView instance.

405

Table 11 – Property origin for boot sources

CIM_PhysicalComputerSystemView property name	CIM_BootSourceSetting property name (origin)
StructuredBootString[]	StructuredBootString
OneTimeBootSource	n/a

406 **7.1.2.9.1 CIM_PhysicalComputerSystemView.StructuredBootString**

407 This property represents the boot sources that are available to be used for the next one-time boot of the
408 physical computer system.

409 **7.1.2.9.2 CIM_PhysicalComputerSystemView.OneTimeBootSource**

410 This property represents the boot source that is used for the next one-time boot of the physical computer
411 system. The value of this property is an index into the StructuredBootString property. A value of NULL
412 shall represent that the one-time boot source is not configured.

413 **7.1.2.10 Representing system identity information**

414 **7.1.2.10.1 CIM_PhysicalComputerSystemView.Dedicated**

415 When implemented according to Table 2, the Dedicated property shall indicate the purposes to which the
416 physical computer system is dedicated, if any, and what functionality is provided. See the specialized
417 profiles of DSP1052 for requirements (for example, DSP1004 and DSP1058).

418 **7.1.2.10.2 CIM_PhysicalComputerSystemView.OtherDedicatedDescriptions**

419 When implemented according to Table 2, the OtherDedicatedDescriptions property shall contain strings
420 describing how or why the physical computer system is dedicated when the Dedicated property includes
421 "Other" (value=2). See the specialized profiles of DSP1052 for requirements (for example, DSP1004 and
422 DSP1058).

423 **7.1.2.11   Representing system FRU information**

424 **7.1.2.11.1   CIM_PhysicalComputerSystemView.FRUInfoSupported**

425 When implemented according to Table 2, the FRUInfoSupported property shall correspond to the value of
426 the CIM_PhysicalAssetCapabilities.FRUInfoSupported property value associated to the CIM_Chassis
427 instance.

428   8   Methods

429 This clause details the requirements for supporting intrinsic operations and extrinsic methods for the CIM
430 elements defined by this profile.

431   8.1   CIM_PhysicalComputerSystemView.RequestStateChange( )

432 When the ImplementedFeatures property of the CIM_RegisteredProfile instance contains
433 "DMTF:ComputerSystemView", RequestStateChange( ) shall be implemented and the implemented
434 method shall not return a value of 1 (Method is unsupported).

435 Invocation of the RequestStateChange( ) method changes the physical computer system's state to the
436 value specified in the RequestedState parameter. A return code value of zero shall indicate that the
437 requested state change was initiated successfully.

438 Detailed requirements of the RequestStateChange( ) method are specified in Table 12 and Table 13.

439 No standard messages are defined.

440 Invoking the RequestStateChange( ) method multiple times could result in earlier requests being
441 overwritten or lost.

442 See CIM_ComputerSystem.RequestStateChange( ) in DSP1052 for additional details.

443 **Table 12 – CIM_PhysicalComputerSystemView.RequestStateChange( ) method: Return code**
444 **values**

Value	Description
0	Request was successfully executed.
2	Error occurred.
4096	Job started: REF returned to started CIM_ConcreteJob.

445

446   **Table 13 – CIM_PhysicalComputerSystemView.RequestStateChange( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Valid state values (mapped to CIM_ComputerSystem.RequestStateChange( )) 2 (Enabled) – On 3 (Disabled) – Off-Soft 9 (Quiesce) – Sleep 11 (Reset) – Power-Cycle (Off-Soft)
IN	TimeoutPeriod	datetime	Client specified maximum amount of time the transition to a new state is supposed to take: 0 or NULL – No time requirements <interval> – Maximum time allowed
OUT	Job	CIM_ConcreteJob REF	Returned if job started

8.2 CIM_PhysicalComputerSystemView.ClearLog( )

448 When the ImplementedFeatures property of the CIM_RegisteredProfile instance contains
449 "DMTF:RecordLogView", ClearLog( ) may be implemented.

450 Invocation of the ClearLog( ) method deletes all the entries of the specified record log of the physical
451 computer system identified by the LogInstanceID parameter. A return code value of zero shall indicate
452 that the clearing of the log entries was initiated successfully.

453 Detailed requirements of the ClearLog( ) method are specified in Table 14 and Table 15.

454 No standard messages are defined.

455   **Table 14 – CIM_PhysicalComputerSystemView.ClearLog( ) method: Return code values**

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred.

456   **Table 15 – CIM_PhysicalComputerSystemView.ClearLog( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	LogInstanceID	String	Identifier of record log that is requested to be cleared

8.3 CIM_PhysicalComputerSystemView.InstallSoftwareFromURI( )

458 When the ImplementedFeatures property of the CIM_RegisteredProfile instance contains
459 "DMTF:SoftwareInventoryView", InstallSoftwareFromURI( ) may be implemented.

460 Invocation of the InstallSoftwareFromURI( ) method starts a job to install software from the designated
461 URI to the physical computer system. A return code value of zero shall indicate that the installation of
462 software was initiated successfully. Based on the payloads, implementations shall determine whether the
463 installation is intended for BIOS or Management Firmware.

464 Detailed requirements of the InstallSoftwareFromURI( ) method are specified in Table 16 and Table 17.

465 No standard messages are defined.

466 See CIM_SoftwareInstallationService.InstallFromURI( ) in DSP1025 for additional details.

467 **Table 16 – CIM_PhysicalComputerSystemView.InstallSoftwareFromURI( ) method: Return code**
468 **values**

Value	Description
0	Job completed with no error.
1	Method is unsupported.
2	Error occurred.
4096	Job started: REF returned to started CIM_ConcreteJob.

469 **Table 17 – CIM_PhysicalComputerSystemView.InstallSoftwareFromURI( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	URI	string	A URI for the software.
IN	InstallOptions[]	uint16	Options to control the install process. See CIM_SoftwareInstallationService.InstallFrom URI( ) in DSP1025 for additional details.
IN	Classifications[]	uint16	Identify the classification of software to install. See CIM_SoftwareIdentity.Classifications in DSP1023 for additional details.
IN	InstallOptionsValues[]	string	InstallOptionsValues is an array of strings providing additional information to InstallOptions for the method to install the software. See CIM_SoftwareInstallationService.InstallFrom URI( ) in DSP1025 for additional details.
OUT	Job	CIM_ConcreteJob REF	Returned if job started

470 8.4 CIM_PhysicalComputerSystemView.SetOneTimeBootSource( )

471 When the ImplementedFeatures property of the CIM_RegisteredProfile instance contains
472 "DMTF:BootControlView", SetOneTImeBootSource( ) may be implemented.

473 Invocation of the SetOneTImeBootSource( ) method sets the boot source for the next boot only. A return
474 code value of zero shall indicate that the new one time boot source was set successfully.

475 If the StructuredBootString parameter contains a value not contained in the StructuredBootString property
476 of the CIM_PhysicalComputerSystemView instance, then the method shall return 2 (Error Occurred).

477 Detailed requirements of the SetOneTimeBootSource( ) method are specified in Table 18 and Table 19.

478 No standard messages are defined.

479   **Table 18 – CIM_PhysicalComputerSystemView.SetOneTimeBootSource( ) method: Return code**
480   **values**

Value	Description
0	Completed with no error.
1	Method is unsupported.
2	Error occurred.
4096	Job started: REF returned to started CIM_ConcreteJob.

481   **Table 19 – CIM_PhysicalComputerSystemView.SetOneTimeBootSource( ) method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	StructuredBootString	string	A StructuredBootString value
OUT	Job	CIM_ConcreteJob REF	Returned if job started

482   8.5   Profile conventions for operations

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

485   The default list of operations is as follows:

486   • GetInstance

487   • EnumerateInstances

488   • EnumerateInstanceNames

489   • Associators

490   • AssociatorNames

491   • References

492   • ReferenceNames

493   8.6   CIM_PhysicalComputerSystemView

494   Table 20 lists operations that either have special requirements beyond those from DSP0200 or shall not
495   be supported.

496   **Table 20 – Operations: CIM_PhysicalComputerSystemView**

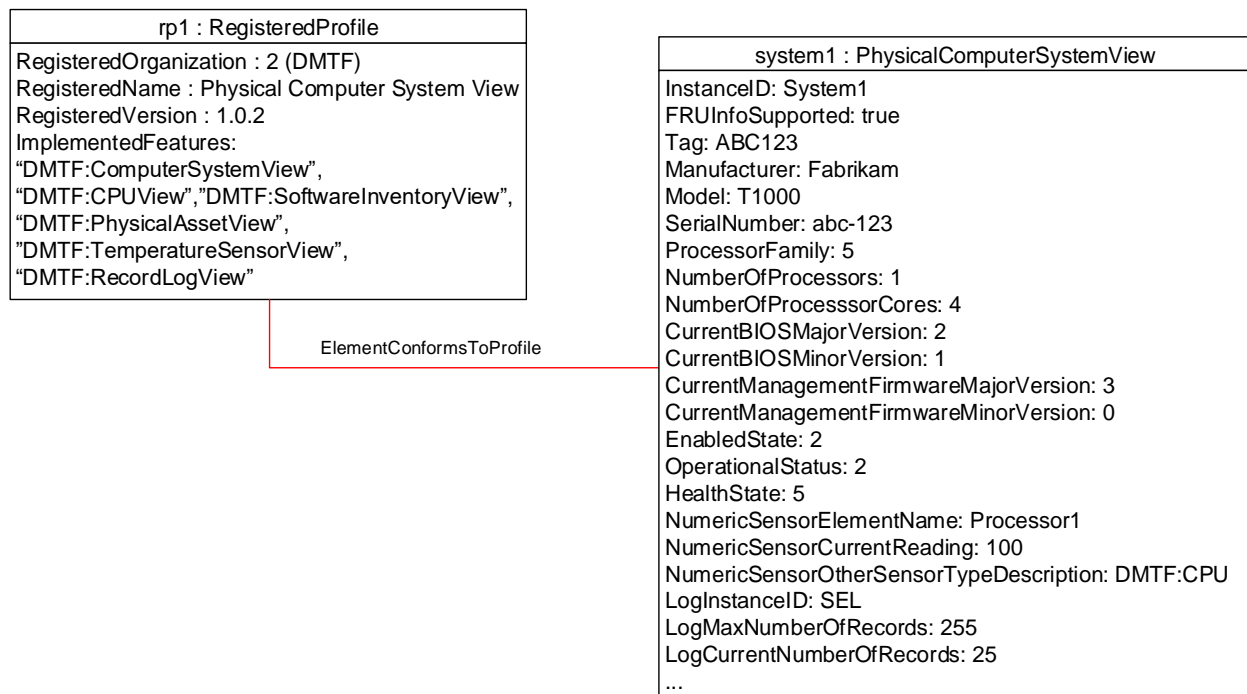| Operation	Requirement	Description
InvokeMethod	Conditional	If "DMTF:ComputerSystemView", "DMTF:RecordLogView", "DMTF:BootControlView", or "DMTF:SoftwareUpdateView" is an implemented feature, this operation shall be supported. See 8.

497   8.7   CIM_ElementView

498   All operations in the default list in 8.5 shall be implemented as defined in DSP0200.

499 # 9 Use cases

500 This clause contains object diagrams and use cases for the *Physical Computer System View Profile*.

501 ## 9.1 Miscellaneous object diagrams

502 The object diagram in Figure 2 shows one possible method for advertising profile conformance. The
503 instances of CIM_RegisteredProfile are used to identify the version of the *Physical Computer System*
504 *View Profile* with which an instance of CIM_PhysicalComputerSystemView and its associated instances
505 are conformant. An instance of CIM_RegisteredProfile exists for each profile that is instrumented in the
506 system. An instance of CIM_RegisteredProfile identifies the "DMTF Physical Computer System View
507 Profile version 1.0.0". This diagram represents a simple managed computer system that only implements
508 the Profile Registration Profile and the Physical Computer System View Profile. The implementation of
509 Physical Computer System View Profile indicates that properties and methods related to Computer
510 System, CPU, Software Inventory, Physical Asset, Sensors, and Record Log are implemented as
511 advertised by using the ImplementedFeatures property of the instance of CIM_RegisteredProfile.

```
┌─────────────────────────────────────────────┐
│          rp1 : RegisteredProfile             │
├─────────────────────────────────────────────┤
│ RegisteredOrganization : 2 (DMTF)            │
│ RegisteredName : Physical Computer System View│
│ RegisteredVersion : 1.0.2                    │
│ ImplementedFeatures:                         │
│ "DMTF:ComputerSystemView",                   │
│ "DMTF:CPUView","DMTF:SoftwareInventoryView", │
│ "DMTF:PhysicalAssetView",                    │
│ "DMTF:TemperatureSensorView",                │
│ "DMTF:RecordLogView"                         │
└─────────────────────────────────────────────┘

┌─────────────────────────────────────────────────────┐
│         system1 : PhysicalComputerSystemView         │
├─────────────────────────────────────────────────────┤
│ InstanceID: System1                                  │
│ FRUInfoSupported: true                               │
│ Tag: ABC123                                          │
│ Manufacturer: Fabrikam                               │
│ Model: T1000                                         │
│ SerialNumber: abc-123                                │
│ ProcessorFamily: 5                                   │
│ NumberOfProcessors: 1                                │
│ NumberOfProcesssorCores: 4                           │
│ CurrentBIOSMajorVersion: 2                           │
│ CurrentBIOSMinorVersion: 1                           │
│ CurrentManagementFirmwareMajorVersion: 3             │
│ CurrentManagementFirmwareMinorVersion: 0             │
│ EnabledState: 2                                      │
│ OperationalStatus: 2                                 │
│ HealthState: 5                                       │
│ NumericSensorElementName: Processor1                 │
│ NumericSensorCurrentReading: 100                     │
│ NumericSensorOtherSensorTypeDescription: DMTF:CPU    │
│ LogInstanceID: SEL                                   │
│ LogMaxNumberOfRecords: 255                           │
│ LogCurrentNumberOfRecords: 25                        │
│ ...                                                  │
└─────────────────────────────────────────────────────┘

ElementConformsToProfile
```

512

513 **Figure 2 – Simple PhysicalComputerSystemView implementation**

514 ## 9.2 Small footprint managed computer system with additional CIM Schema
515 ## implemented

516 The object diagram in Figure 3 shows a small footprint managed computer system that implements the
517 same features as the managed computer system in Figure 2, but also implements CIM Schema that is
518 associated to specific properties. This managed computer system does not claim conformance to
519 additional profiles, but exposes additional properties/methods through the CIM classes associated to the
520 instance of CIM_PhysicalComputerSystemView.
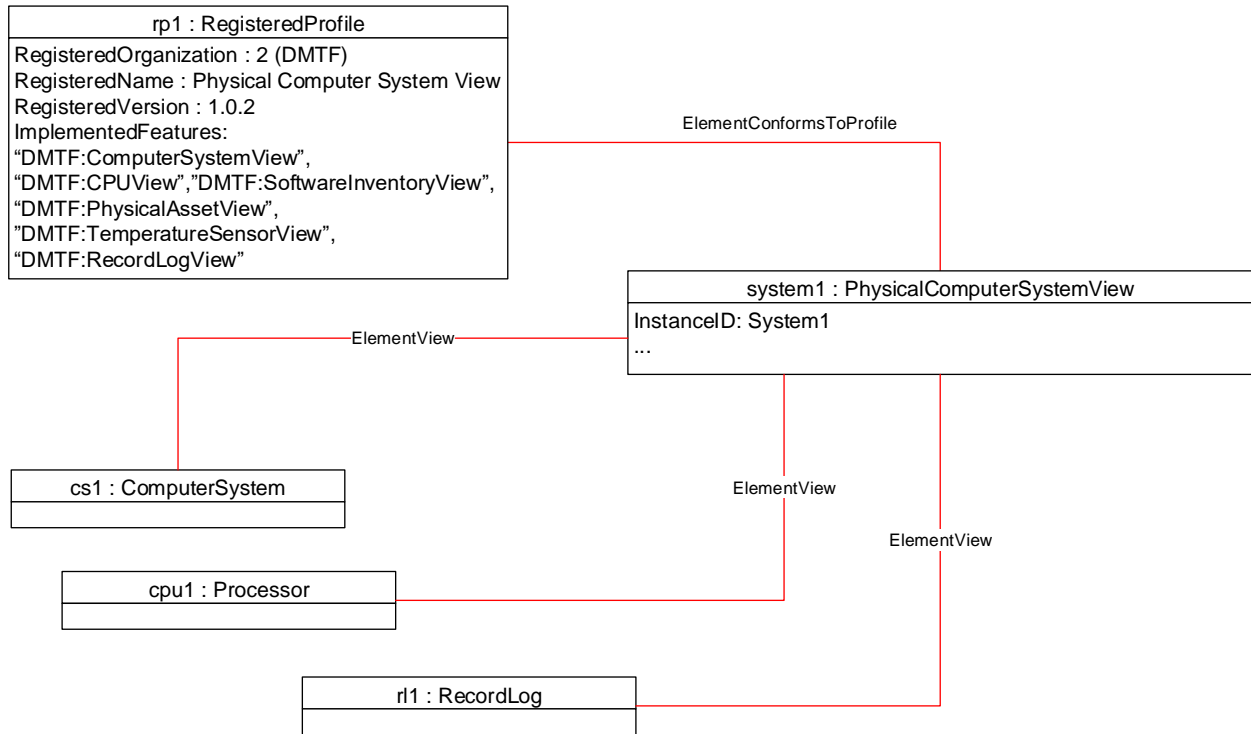
522   **Figure 3 – Small footprint PhysicalComputerSystemView implementation**

523   **9.3   Large managed computer system with additional CIM Profiles implemented**

524   The object diagram in Figure 4 shows a large managed computer system that implements the same
525   features as the managed computer system in Figure 3, but also implements CIM Schema that is
526   conformant to CIM Profiles providing richer management capabilities.
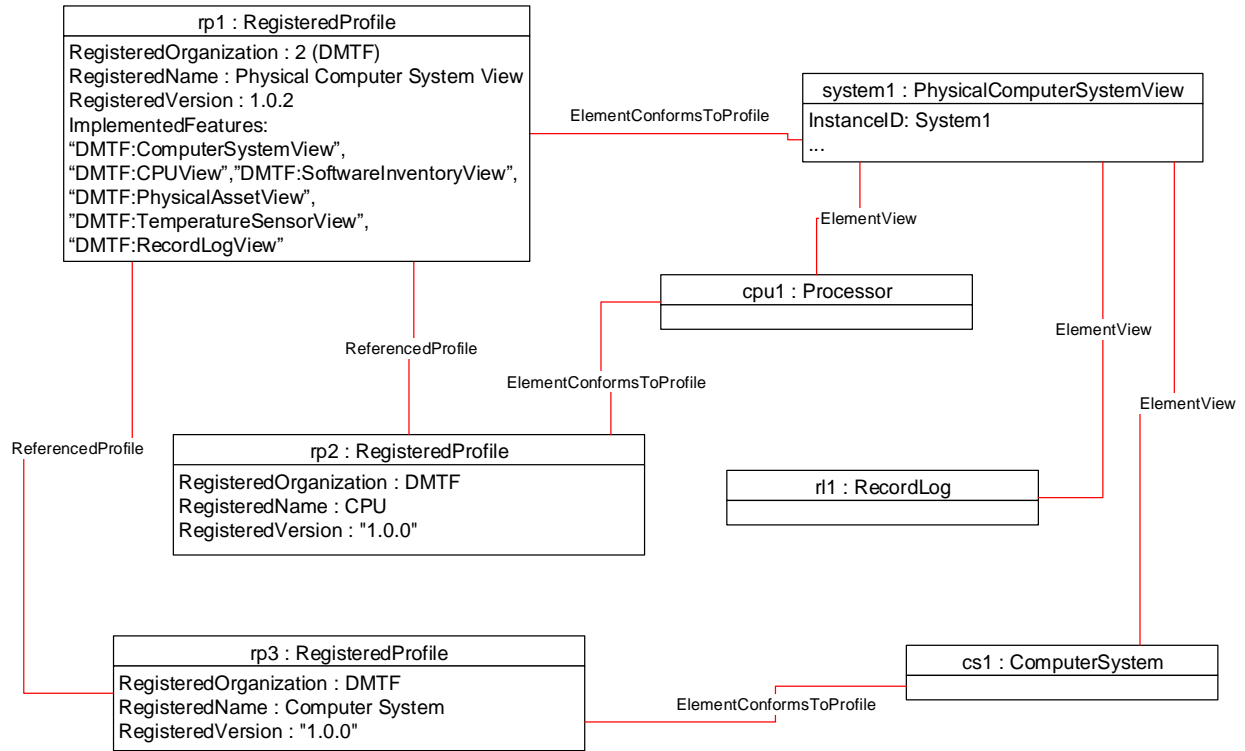
527

528 **Figure 4 – Rich PhysicalComputerSystemView implementation**

9.4 Managed system exposing multiple temperature sensors

530 The object diagram in Figure 5 shows a managed computer system that implements multiple temperature
531 sensors for Inlet, CPU, and Base Board. Although the instances of CIM_NumericSensor and
532 corresponding association are not required, they illustrate how to represent multiple numeric sensors as
533 an indexed array in the CIM_PhysicalComputerSystemView instance.
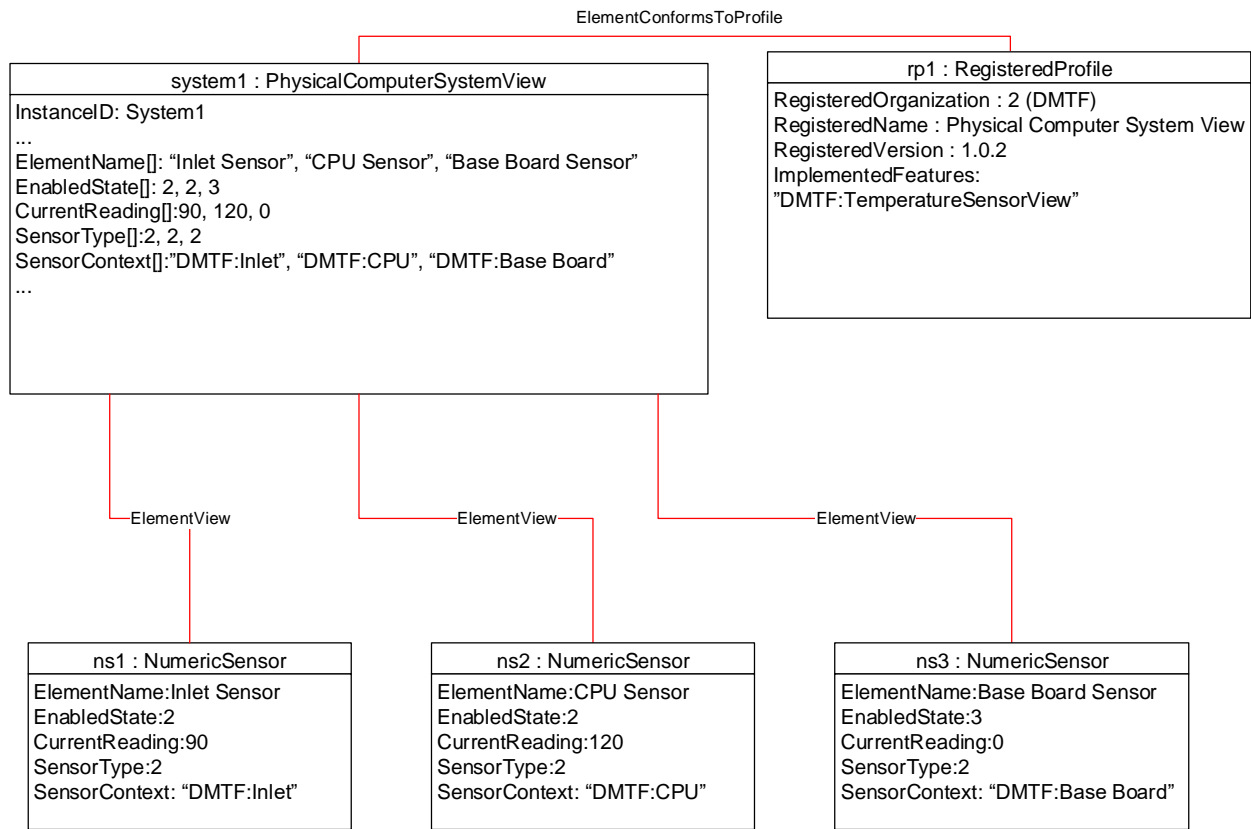
534

535   **Figure 5 – PhysicalComputerSystemView implementation of multiple temperature sensors**

536 ## 9.5 Determine the enabled state of the physical computer system

537 To determine the enabled state:

538   1) Verify that the managed computer system implements the DMTF:ComputerSystemView feature
539   by reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

540   2) Read the EnabledState property of the instance of CIM_PhysicalComputerSystemView.

541 ## 9.6 Change the enabled state of the physical computer system

542 To change the enabled state:

543   1) Verify that the managed computer system implements the DMTF:ComputerSystemView feature
544   by reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

545   2) Invoke the RequestStateChange( ) method of the instance of CIM_PhysicalComputerSystemView
546   providing the new requested state.

547   3) See Table 13 for the supported list of requested enabled states.

548 ## 9.7 Get properties of a specific record log of the physical computer system

549 To get the properties of a specific record log:

550   1) Verify that the managed computer system implements the DMTF:RecordLogView feature by
551   reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

552   2) Identify the index to the specific log that matches the LogInstanceID.

553   3) Use this index to read the ordered arrays MaxNumberOfRecords, CurrentNumberOfRecords,
554   OverWritePolicy, and LogState for that specific record log.

555 ## 9.8 Browse the records of a log of the physical computer system

556 To browse log entries:

557   1) Verify that the managed computer system implements the DMTF:RecordLogView feature by
558   reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

559   2) Verify that an instance of CIM_RegisteredProfile for Record Log Profile exists.

560   3) Traverse the association to the CIM_RecordLog instance with the LogInstanceID by using the
561   CIM_ElementView association class.

562   4) Enumerate instances of CIM_LogEntry that are associated through the CIM_LogManagesRecord
563   association to the given instance of CIM_RecordLog.
564

565   ## 9.9   Monitor temperature sensor readings of the physical computer system

566   To monitor CPU temperature sensor readings:

567   1)   Verify that the managed computer system implements the DMTF:NumericSensorView feature by
568   reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

569   2)   Identify the index to a specific sensor that contains "DMTF:CPU" as the value for
570   NumericSensorContext.

571   3)   Use this index to read the ordered arrays NumericSensorBaseUnits, NumericSensorUnitModifier,
572   NumericSensorRateUnits, and NumericSensorCurrentReading to compute the reading value for
573   that specific sensor.

574   ## 9.10   Configure a source of the physical computer system for next reboot only

575   To configure the boot source for next reboot only:

576   1)   Verify that the managed computer system implements the DMTF:BootControlView feature by
577   reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

578   2)   Invoke the SetOneTimeBootSource( ) method of the instance of
579   CIM_PhysicalComputerSystemView to reflect the desired boot source based on an index value
580   from the StructedBootString array.

581   ## 9.11   Update the BIOS firmware of the physical computer system

582   To update the BIOS firmware:

583   1)   Verify that the managed computer system implements the DMTF:SoftwareUpdateView feature by
584   reading the ImplementedFeatures property of the CIM_RegisteredProfile instance.

585   2)   Invoke the InstallSoftwareFromURI( ) method of the instance of
586   CIM_PhysicalComputerSystemView providing the classification value of 6 (Firmware/BIOS),
587   location of the software, and required options.
588

589 # 10 CIM Elements

590 Table 21 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be
591 implemented as described in Table 21.

592 **Table 21 – CIM Elements: Physical Computer System View Profile**

Element Name	Requirement	Description
Classes		
CIM_PhysicalComputerSystemView	Mandatory	See 10.1.
CIM_RegisteredProfile	Mandatory	See 10.2.
CIM_ElementView	Optional	See 10.3
CIM_ElementConformsToProfile	Mandatory	See 10.4
Indications		
None defined in this profile		

593 ## 10.1 CIM_PhysicalComputerSystemView

594 CIM_PhysicalComputerSystemView describes a physical computer system and associated management
595 information in a managed computer system.

596 **Table 22 – Class: CIM_PhysicalComputerSystemView**

Elements	Requirement	Description
InstanceID	Mandatory	Key. See 7.1.2.1.
OtherIdentifyingInfo	Conditional	See Table 2.
IdentifyingDescriptions	Conditional	See Table 2.
EnabledState	Mandatory	See Table 2.
RequestedState	Conditional	See Table 2.
OperationalStatus	Mandatory	See Table 2.
HealthState	Mandatory	See Table 2.
Dedicated	Mandatory	See Table 2.
OtherDedicatedDescriptions	Conditional	See Table 2.
FRUInfoSupported	Conditional	See Table 2.
Tag	Conditional	See Table 2.
Manufacturer	Conditional	See Table 2.
Model	Conditional	See Table 2.
SKU	Conditional	See Table 2.
SerialNumber	Conditional	See Table 2.
Version	Conditional	See Table 2.

Elements	Requirement	Description
PartNumber	Conditional	See Table 2.
PowerUtilizationModesSupported	Conditional	See Table 2.
PowerUtilizationMode	Conditional	See Table 2.
PowerAllocationLimit	Conditional	See Table 2.
NumericSensorElementName	Conditional	See Table 2.
NumericSensorEnabledState	Conditional	See Table 2.
NumercSensorHealthState	Conditional	See Table 2.
NumericSensorCurrentState	Conditional	See Table 2.
NumericSensorPrimaryStatus	Conditional	See Table 2.
NumericSensorBaseUnits	Conditional	See Table 2.
NumericSensorUnitModifier	Conditional	See Table 2.
NumericSensorRateUnits	Conditional	See Table 2.
NumericSensorCurrentReading	Conditional	See Table 2.
NumericSensorSensorType	Conditional	See Table 2.
NumericSensorOtherSensorTypeDescription	Conditional	See Table 2.
NumericSensorContext	Conditional	See Table 2.
NumericSensorUpperThresholdNonCritical	Conditional	See Table 2.
NumericSensorUpperThresholdCritical	Conditional	See Table 2.
NumericSensorUpperThresholdFatal	Conditional	See Table 2.
NumericSensorLowerThresholdNonCritical	Conditional	See Table 2.
NumericSensorLowerThresholdCritical	Conditional	See Table 2.
NumericSensorLowerThresholdFatal	Conditional	See Table 2.
LogInstanceID	Conditional	See Table 2.
LogMaxNumberOfRecords	Conditional	See Table 2.
LogCurrentNumberOfRecords	Conditional	See Table 2.
LogOverWritePolicy	Conditional	See Table 2.
LogState	Conditional	See Table 2.
StructuredBootString	Conditional	See Table 2.
OneTimeBootSource	Conditional	See Table 2.
NumberOfProcessors	Conditional	See Table 2.
NumberOfProcessorCores	Conditional	See Table 2.
NumberOfProcessorThreads	Conditional	See Table 2.
ProcessorFamily	Conditional	See Table 2.
ProcessorCurrentClockSpeed	Conditional	See Table 2.
ProcessorMaxClockSpeed	Conditional	See Table 2.

Elements	Requirement	Description
MemoryBlockSize	Conditional	See Table 2.
MemoryNumberOfBlocks	Conditional	See Table 2.
MemoryConsumableBlocks	Conditional	See Table 2.
CurrentBIOSMajorVersion	Conditional	See Table 2.
CurrentBIOSMinorVersion	Conditional	See Table 2.
CurrentBIOSRevisionNumber	Conditional	See Table 2.
CurrentBIOSBuildNumber	Conditional	See Table 2.
CurrentBIOSVersionString	Conditional	See Table 2.
CurrentManagementFirmwareMajorVersion	Conditional	See Table 2.
CurrentManagementFirmwareMinorVersion	Conditional	See Table 2.
CurrentManagementFirmwareRevisionNumber	Conditional	See Table 2.
CurrentManagementFirmwareBuildNumber	Conditional	See Table 2.
CurrentManagementFirmwareElementName	Conditional	See Table 2.
CurrentManagementFirmwareVersionString	Conditional	See Table 2.
OSType	Conditional	See Table 2.
OSVersion	Conditional	See Table 2.
OSEnabledState	Conditional	See Table 2.
RequestStateChange( )	Mandatory	See 8.1.
ClearLog( )	Conditional	See 8.2.
InstallSoftwareFromURI( )	Conditional	See 8.3.
SetOneTimeBootSource( )	Conditional	See 8.4.

10.2 CIM_RegisteredProfile

CIM_RegisteredProfile identifies the *Physical Computer System View Profile* in order for a client to
determine whether an instance of CIM_PhysicalComputerSystemView is conformant with this profile. The
CIM_RegisteredProfile class is defined by the *Profile Registration Profile* (DSP1033). With the exception
of the mandatory values specified for the properties in Table 23, the behavior of the
CIM_RegisteredProfile instance is in accordance with DSP1033.

Table 23 – Class: CIM_RegisteredProfile

Elements	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of "Physical Computer System View".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.3".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

Elements	Requirement	Description
ImplementedFeatures	Mandatory	This property shall contain "DMTF:ComputerSystemView". This property may contain one or more of these values: "DMTF:RecordLogView", "DMTF:NumericSensorView", "DMTF:CPUView", "DMTF:BootControlView", "DMTF:SoftwareInventoryView", "DMTF:PhysicalAssetView", "DMTF:SystemMemoryView", "DMTF:PowerUtilizationManagementView", "DMTF:OSView", "DMTF:SoftwareUpdateView". Presences of values in this property only indicate implemented properties/methods in CIM_PhysicalComputerSystemView and do not indicate conformance to additional CIM Profiles.

604 10.3 CIM_ElementView

605 CIM_ElementView associates additional CIM Schema to the CIM_PhysicalComputerSystemView
606 instance.

607   **Table 24 – Class: CIM_ElementView**

Elements	Requirement	Description
Antecedent	Mandatory	This property shall be a reference to the CIM_ManagedElement that is an instance in the normalized representation of the managed resource.
Dependent	Mandatory	This property shall be a reference to the CIM_PhysicalComputerSystemView instance.

608 10.4 CIM_ElementConformsToProfile

609 CIM_ElementConformsToProfile associates the instance of CIM_RegisteredProfile to the
610 CIM_PhysicalComputerSystemView instance.

611   **Table 25 – Class: CIM_ElementConformsToProfile**

Elements	Requirement	Description
ConformantStandard	Mandatory	This property shall be a reference to the CIM_RegisteredProfile instance for the Physical Computer System View Profile.
ManagedElement	Mandatory	This property shall be a reference to the CIM_PhysicalComputerSystemView instance.

612   **ANNEX A**
613   **(informative)**
614   **Change log**

Version	Date	Description
1.0.0	2013-08-22	
1.0.1	2014-05-22	This errata addresses these issues: • Added clarifying language that RequestStateChange() shall not return 1 (Method is unsupported) as it is mandatory to be implemented. In addition, removed return value 1 (Method is unsupported) from Table 12 • Clarified language that all other methods may be implemented and not conditional as requirements are based on referenced profiles • Corrected language of use cases referring to power state to enabled state • Fixed incorrect table reference in 9.6
1.0.2	2017-03-18	This errata addresses these issues: • Updated RegisteredVersion to reflect errata version number in Section 10.2, and • Updated RegisteredOrganization description to reflect correct value for DMTF in Section 10.2. • Updated use cases to reflect the above fixes.
1.0.3	2022-01-01	This errata addresses these issues: • Reference to added Profile Registration Profile 1.1 • Profile Registration Profile 1.1 added to referenced profiles.

615