



1  
2  
3  
4  
5

**Document Number: DSP1009**

**Date: 2009-10-28**

**Version: 1.0.2**

## 6 **Sensors Profile**

7 **Document Type: Specification**  
8 **Document Status: DMTF Standard**  
9 **Document Language: E**

## 10 Copyright Notice

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

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

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27 withdrawn or modified after publication, and shall be indemnified and held harmless by any party  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

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

# CONTENTS

34	Foreword .....	5
35	Introduction .....	6
36	1 Scope .....	7
37	2 Normative References.....	7
38	2.1 Approved References .....	7
39	2.2 Other References.....	7
40	3 Terms and Definitions .....	7
41	4 Symbols and Abbreviated Terms .....	8
42	5 Synopsis .....	9
43	6 Description .....	9
44	7 Implementation Requirements .....	10
45	7.1 CIM_Sensor and CIM_NumericSensor .....	10
46	7.2 CIM_Sensor.PossibleStates .....	10
47	7.3 CIM_NumericSensor.PossibleStates.....	10
48	7.4 CIM_Sensor.CurrentState and CIM_NumericSensor.CurrentState .....	11
49	7.5 CIM_NumericSensor.LowerThresholdNonCritical.....	11
50	7.6 CIM_NumericSensor.UpperThresholdNonCritical.....	11
51	7.7 CIM_NumericSensor.LowerThresholdCritical .....	11
52	7.8 CIM_NumericSensor.UpperThresholdCritical .....	12
53	7.9 CIM_NumericSensor.LowerThresholdFatal .....	12
54	7.10 CIM_NumericSensor.UpperThresholdFatal .....	12
55	7.11 CIM_NumericSensor.SupportedThresholds.....	12
56	7.12 CIM_NumericSensor.SettableThresholds .....	12
57	7.13 CIM_EnabledLogicalElementCapabilities.....	12
58	7.14 Sensor State Management .....	13
59	7.15 CIM_Sensor.RequestedState and CIM_NumericSensor.RequestedState .....	13
60	7.16 CIM_Sensor.EnabledState and CIM_NumericSensor.EnabledState.....	14
61	7.17 CIM_Sensor.OtherSensorTypeDescription and	
62	CIM_NumericSensor.OtherSensorTypeDescription.....	14
63	7.18 CIM_SystemDevice and CIM_AssociatedSensor .....	14
64	7.19 CIM_Sensor.ElementName and CIM_NumericSensor.ElementName .....	14
65	8 Methods.....	15
66	8.1 CIM_Sensor.RequestStateChange( ) and CIM_NumericSensor.RequestStateChange( ) .....	15
67	8.2 CIM_NumericSensor.RestoreDefaultThresholds( ) .....	15
68	8.3 Profile Conventions for Operations.....	16
69	8.4 CIM_Sensor .....	16
70	8.5 CIM_NumericSensor.....	17
71	8.6 CIM_EnabledLogicalElementCapabilities.....	18
72	8.7 CIM_ElementCapabilities .....	18
73	8.8 CIM_SystemDevice .....	18
74	8.9 CIM_AssociatedSensor .....	19
75	9 Use Cases.....	19
76	9.1 Object Diagrams .....	19
77	9.2 Show All Current States of the Monitored Devices in the Computer System.....	20
78	9.3 Find Sensors Associated with a Specific Component .....	20
79	9.4 Change Upper Non-Critical Threshold of Numeric Sensor.....	20

80	9.5	Reset Sensor to Threshold Defaults .....	20
81	9.6	Determining If ElementName Can Be Modified .....	21
82	9.7	Determining If State Management Is Supported .....	21
83	10	CIM Elements .....	21
84	10.1	CIM_Sensor .....	22
85	10.2	CIM_NumericSensor .....	23
86	10.3	CIM_EnabledLogicalElementCapabilities .....	24
87	10.4	CIM_ElementCapabilities .....	24
88	10.5	CIM_SystemDevice .....	24
89	10.6	CIM_AssociatedSensor .....	25
90	10.7	CIM_RegisteredProfile .....	25
91		ANNEX A (informative) Change Log .....	26

92

## 93 Figures

94	Figure 1 – Sensors Profile: Class Diagram .....	9
95	Figure 2 – Sensors Profile: Object Diagram .....	20
96		

## 97 Tables

98	Table 1 – Related Profiles .....	9
99	Table 2 – CIM_Sensor.PossibleStates Values for Discrete Sensors .....	10
100	Table 3 – CIM_NumericSensor.PossibleStates Values for Numeric Sensors .....	11
101	Table 4 – EnabledState Value Description .....	14
102	Table 5 – RequestStateChange( ) Method: Return Code Values .....	15
103	Table 6 – RequestStateChange( ) Method: Parameters .....	15
104	Table 7 – CIM_NumericSensor.RestoreDefaultThresholds( ) Method: Return Code Values .....	16
105	Table 8 – Operations: CIM_Sensor .....	16
106	Table 9 – Operations: CIM_NumericSensor .....	17
107	Table 10 – Operations: CIM_ElementCapabilities .....	18
108	Table 11 – Operations: CIM_SystemDevice .....	18
109	Table 12 – Operations: CIM_AssociatedSensor .....	19
110	Table 13 – CIM Elements: Sensors Profile Profile .....	21
111	Table 14 – Class: CIM_Sensor .....	22
112	Table 15 – Class: CIM_NumericSensor .....	23
113	Table 16 – Class: CIM_EnabledLogicalElementCapabilities .....	24
114	Table 17 – Class: CIM_ElementCapabilities .....	24
115	Table 18 – Class: CIM_SystemDevice .....	24
116	Table 19 – Class: CIM_AssociatedSensor .....	25
117	Table 20 – Class: CIM_RegisteredProfile .....	25
118		

119

## Foreword

120 The *Sensors Profile* (DSP1009) was prepared by the Server Management Working Group of the DMTF.

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

### 123 **Acknowledgments**

124 The authors wish to acknowledge the following people.

#### 125 **Editors:**

- 126 • Hemal Shah – Broadcom Corporation
- 127 • Jon Hass – Dell Inc.
- 128 • Khachatur Papanyan – Dell Inc.
- 129 • Jim Davis (Errata) – WBEM Solutions

#### 130 **Contributors:**

- 131 • Hemal Shah – Broadcom Corporation
- 132 • Jon Hass – Dell Inc.
- 133 • Khachatur Papanyan – Dell Inc.
- 134 • Linda Martinez – Dell Inc.
- 135 • Jeff Hilland – HP
- 136 • Christina Shaw – HP
- 137 • Aaron Merkin – IBM
- 138 • Jeff Lynch – IBM
- 139 • Perry Vincent – Intel
- 140 • John Leung – Intel
- 141 • Jim Davis – WBEM Solutions

142

143

## Introduction

144 This document defines the classes used to describe the sensors in a managed system. The document  
145 also defines association classes that describe the relationship of the sensors with the monitored devices  
146 and with DMTF profile version information. The information in this specification is intended to be sufficient  
147 for a provider or consumer of this data to identify unambiguously the classes, properties, methods, and  
148 values that are mandatory to be instantiated and manipulated to represent and manage sensors of  
149 managed systems and subsystems that are modeled using the DMTF CIM core and extended model  
150 definitions.

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

153

# Sensors Profile

## 154 1 Scope

155 The *Sensors Profile* extends the management capabilities of referencing profiles by adding the capability  
156 to represent sensors. The sensor's relationship with devices and the profile's registration for the schema  
157 implementation version information are also described.

## 158 2 Normative References

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

### 162 2.1 Approved References

163 DMTF DSP0004, *CIM Infrastructure Specification 2.5*,  
164 [http://www.dmtf.org/standards/published\\_documents/DSP0004\\_2.5.pdf](http://www.dmtf.org/standards/published_documents/DSP0004_2.5.pdf)

165 DMTF DSP0200, *CIM Operations over HTTP 1.3*,  
166 [http://www.dmtf.org/standards/published\\_documents/DSP0200\\_1.3.pdf](http://www.dmtf.org/standards/published_documents/DSP0200_1.3.pdf)

167 DMTF DSP1001, *Management Profile Specification Usage Guide 1.0*,  
168 [http://www.dmtf.org/standards/published\\_documents/DSP1001\\_1.0.pdf](http://www.dmtf.org/standards/published_documents/DSP1001_1.0.pdf)

169 DMTF DSP1033, *Profile Registration Profile 1.0*,  
170 [http://www.dmtf.org/standards/published\\_documents/DSP1033\\_1.0.pdf](http://www.dmtf.org/standards/published_documents/DSP1033_1.0.pdf)

### 171 2.2 Other References

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

## 174 3 Terms and Definitions

175 For the purposes of this document, the following terms and definitions apply.

### 176 3.1

#### 177 can

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

### 179 3.2

#### 180 cannot

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

### 182 3.3

#### 183 conditional

184 indicates requirements to be followed strictly to conform to the document when the specified conditions  
185 are met

- 186 **3.4**  
187 **mandatory**  
188 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
189 permitted
- 190 **3.5**  
191 **may**  
192 indicates a course of action permissible within the limits of the document
- 193 **3.6**  
194 **need not**  
195 indicates a course of action permissible within the limits of the document
- 196 **3.7**  
197 **optional**  
198 indicates a course of action permissible within the limits of the document
- 199 **3.8**  
200 **referencing profile**  
201 indicates a profile that owns the definition of this class and can include a reference to this profile in its  
202 "Related Profiles" table
- 203 **3.9**  
204 **shall**  
205 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
206 permitted
- 207 **3.10**  
208 **shall not**  
209 indicates requirements to be followed strictly to conform to the document and from which no deviation is  
210 permitted
- 211 **3.11**  
212 **should**  
213 indicates that among several possibilities, one is recommended as particularly suitable, without  
214 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 215 **3.12**  
216 **should not**  
217 indicates that a certain possibility or course of action is deprecated but not prohibited
- 218 **4 Symbols and Abbreviated Terms**  
219 None.



## 220 5 Synopsis

221 **Profile Name:** Sensors

222 **Version:** 1.0.2

223 **Organization:** DMTF

224 **CIM Schema Version:** 2.19.1

225 **Central Class:** CIM\_Sensor

226 **Scoping Class:** CIM\_ComputerSystem

227 The *Sensors Profile* extends the management capability of the referencing profiles by adding the  
 228 capability to represent sensors in a managed system.

229 Table 1 identifies profiles related to this profile.

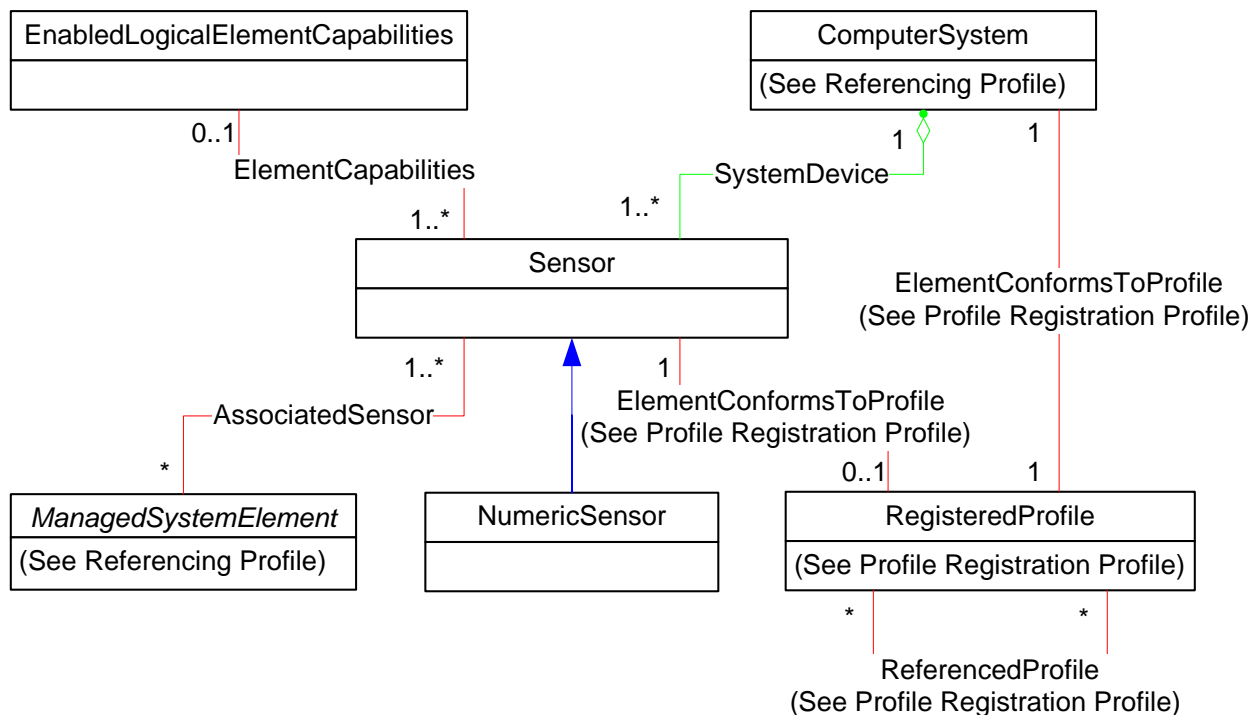
230 **Table 1 – Related Profiles**

Profile Name	Organization	Version	Requirement	Description
Profile Registration	DMTF	1.0	Mandatory	None

## 231 6 Description

232 The *Sensors Profile* describes the properties and methods of sensors in a managed system.

233 Figure 1 represents the class schema for the *Sensors Profile*. For simplicity, the prefix CIM\_ has been  
 234 removed from the names of the classes.



235

236

**Figure 1 – Sensors Profile: Class Diagram**

## 237 7 Implementation Requirements

238 This section details the requirements related to the instantiation of instances and their properties for  
 239 implementations of this profile. The requirements for the implementation of the methods are listed in 8,  
 240 "Methods".

### 241 7.1 CIM\_Sensor and CIM\_NumericSensor

242 The implementation shall instantiate an instance of CIM\_Sensor, including its subclass  
 243 CIM\_NumericSensor.

### 244 7.2 CIM\_Sensor.PossibleStates

245 The CIM\_Sensor.PossibleStates property shall represent an array of the possible string outputs of the  
 246 sensor provided as a value of the CIM\_Sensor.CurrentState property. The CIM\_Sensor.SensorType  
 247 property shall determine which CIM\_Sensor.PossibleStates enumeration set to use.

248 The mappings between the CIM\_Sensor.SensorType property values and the  
 249 CIM\_Sensor.PossibleStates property values are shown in Table 2. When the value of the  
 250 CIM\_Sensor.SensorType property matches a value in the "CIM\_Sensor.SensorType" column of Table 2,  
 251 the CIM\_Sensor.PossibleStates property shall contain an array of values or an array of the subset of  
 252 values specified in the corresponding "CIM\_Sensor.PossibleStates" column. If the value of the  
 253 CIM\_Sensor.SensorType property is not listed in Table 2, the CIM\_Sensor.PossibleStates property shall  
 254 be defined by the implementation. The mapping between the values of CIM\_Sensor.PossibleStates in  
 255 Table 2 and the actual condition of the monitored device is implementation specific.

256 **Table 2 – CIM\_Sensor.PossibleStates Values for Discrete Sensors**

CIM_Sensor.SensorType	CIM_Sensor.PossibleStates
2 (Temperature)	"Bad", "Good", "Unknown"
3 (Voltage)	"Bad", "Good", "Unknown"
4 (Current)	"Bad", "Good", "Unknown"
5 (Tachometer)	"Bad", "Good", "Unknown"
7 (Switch)	"Closed", "Open", "Unknown"
8 (Lock)	"Locked", "Unlocked", "Unknown"
9 (Humidity)	"Humid", "Normal", "Unknown"
10 (Smoke Detection)	"Smoky", "Normal", "Unknown"
11 (Presence)	"Not Present", "Present", "Unknown"
12 (Air Flow)	"Bad", "Good", "Unknown"
13 (Power Consumption)	"Bad", "Good", "Unknown"
14 (Power Production)	"Bad", "Good", "Unknown"
15 (Pressure)	"Bad", "Good", "Unknown"

### 257 7.3 CIM\_NumericSensor.PossibleStates

258 The CIM\_NumericSensor.PossibleStates property shall represent an array of the possible string outputs  
 259 of the sensor provided as a value of the CIM\_NumericSensor.CurrentState property. The  
 260 CIM\_NumericSensor.SensorType property shall determine which CIM\_NumericSensor.PossibleStates  
 261 enumeration set to use.

262 The mappings between the CIM\_NumericSensor.SensorType property values and the  
 263 CIM\_NumericSensor.PossibleStates property values are shown in Table 3. When the value of the  
 264 CIM\_NumericSensor.SensorType property matches a value in the “CIM\_NumericSensor.SensorType”  
 265 column of Table 3, the CIM\_NumericSensor.PossibleStates property shall contain an array of values or  
 266 an array of the subset of the values specified in the corresponding “CIM\_NumericSensor.PossibleStates”  
 267 column. If the value of the CIM\_NumericSensor.SensorType property is not listed in Table 3, the  
 268 CIM\_NumericSensor.PossibleStates property shall be defined by the implementation. The mapping  
 269 between the values of CIM\_NumericSensor.PossibleStates in Table 3 and the actual condition of the  
 270 monitored device is implementation specific.

271 **Table 3 – CIM\_NumericSensor.PossibleStates Values for Numeric Sensors**

CIM_NumericSensor.SensorType	CIM_NumericSensor.PossibleStates
2 (Temperature)	“Non-Critical”, “Lower Non-Critical”, “Upper Non-Critical”, “Critical”, “Lower Critical”, “Upper Critical”, “Fatal”, “Lower Fatal”, “Upper Fatal”, “Normal”, “Unknown”
3 (Voltage)	
4 (Current)	
5 (Tachometer)	
9 (Humidity)	
10 (Smoke Detection)	
12 (Air Flow)	
13 (Power Consumption)	
14 (Power Production)	
15 (Pressure)	

272 **7.4 CIM\_Sensor.CurrentState and CIM\_NumericSensor.CurrentState**

273 The CIM\_Sensor.CurrentState property shall have a value of one of the elements in the  
 274 CIM\_Sensor.PossibleStates array.

275 The CIM\_NumericSensor.CurrentState property shall have a value of one of the elements in the  
 276 CIM\_NumericSensor.PossibleStates array.

277 **7.5 CIM\_NumericSensor.LowerThresholdNonCritical**

278 The CIM\_NumericSensor.LowerThresholdNonCritical property shall be mandatory when the  
 279 CIM\_NumericSensor.SupportedThresholds array contains a value of 0 (LowerThresholdNonCritical).

280 The CIM\_NumericSensor.LowerThresholdNonCritical property shall be settable only if the  
 281 CIM\_NumericSensor.SettableThresholds array contains a value of 0 (LowerThresholdNonCritical).

282 **7.6 CIM\_NumericSensor.UpperThresholdNonCritical**

283 The CIM\_NumericSensor.UpperThresholdNonCritical property shall be mandatory when the  
 284 CIM\_NumericSensor.SupportedThresholds array contains a value of 1 (UpperThresholdNonCritical).

285 The CIM\_NumericSensor.UpperThresholdNonCritical property shall be settable only if the  
 286 CIM\_NumericSensor.SettableThresholds array contains a value of 1 (UpperThresholdNonCritical).

287 **7.7 CIM\_NumericSensor.LowerThresholdCritical**

288 The CIM\_NumericSensor.LowerThresholdCritical property shall be mandatory when the  
 289 CIM\_NumericSensor.SupportedThresholds array contains a value of 2 (LowerThresholdCritical).

290 The CIM\_NumericSensor.LowerThresholdCritical property shall be settable only if the  
291 CIM\_NumericSensor.SettableThresholds array contains a value of 2 (LowerThresholdCritical).

### 292 **7.8 CIM\_NumericSensor.UpperThresholdCritical**

293 The CIM\_NumericSensor.UpperThresholdCritical property shall be mandatory when the  
294 CIM\_NumericSensor.SupportedThresholds array contains a value of 3 (UpperThresholdCritical).

295 The CIM\_NumericSensor.UpperThresholdCritical property shall be settable only if the  
296 CIM\_NumericSensor.SettableThresholds array contains a value of 3 (UpperThresholdCritical).

### 297 **7.9 CIM\_NumericSensor.LowerThresholdFatal**

298 The CIM\_NumericSensor.LowerThresholdFatal property shall be mandatory when the  
299 CIM\_NumericSensor.SupportedThresholds array contains a value of 4 (LowerThresholdFatal).

300 The CIM\_NumericSensor.LowerThresholdFatal property shall be settable only if the  
301 CIM\_NumericSensor.SettableThresholds array contains a value of 4 (LowerThresholdFatal).

### 302 **7.10 CIM\_NumericSensor.UpperThresholdFatal**

303 The CIM\_NumericSensor.UpperThresholdFatal property shall be mandatory when the  
304 CIM\_NumericSensor.SupportedThresholds array contains a value of 5 (UpperThresholdFatal).

305 The CIM\_NumericSensor.UpperThresholdFatal property shall be settable only if the  
306 CIM\_NumericSensor.SettableThresholds array contains a value of 5 (UpperThresholdFatal).

### 307 **7.11 CIM\_NumericSensor.SupportedThresholds**

308 The CIM\_NumericSensor.SupportedThresholds property is an array that contains the list of the  
309 implemented thresholds: LowerThresholdNonCritical, UpperThresholdNonCritical,  
310 LowerThresholdCritical, UpperThresholdCritical, LowerThresholdFatal, and UpperThresholdFatal. When  
311 the implementation does not support any of these threshold properties, the  
312 CIM\_NumericSensor.SupportedThresholds property shall be an empty array.

### 313 **7.12 CIM\_NumericSensor.SettableThresholds**

314 The CIM\_NumericSensor.SettableThresholds property is an array that contains the list of the settable  
315 implemented thresholds: LowerThresholdNonCritical, UpperThresholdNonCritical,  
316 LowerThresholdCritical, UpperThresholdCritical, LowerThresholdFatal, and UpperThresholdFatal. The  
317 CIM\_NumericSensor.SettableThresholds array shall contain the subset of values in the  
318 CIM\_NumericSensor.SupportedThresholds array (see 7.11). When the implementation does not support  
319 any of the settable threshold properties, the CIM\_NumericSensor.SettableThresholds property shall be an  
320 empty array.

### 321 **7.13 CIM\_EnabledLogicalElementCapabilities**

322 When the CIM\_EnabledLogicalElementCapabilities class is instantiated, the instance of  
323 CIM\_EnabledLogicalElementCapabilities shall be associated with the Central Instance through the  
324 CIM\_ElementCapabilities association and used for advertising the capabilities of the Central Instance.

325 At most only one instance of CIM\_EnabledLogicalElementCapabilities shall be associated with a given  
326 instance of CIM\_Sensor or CIM\_NumericSensor.

### 327 **7.13.1 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported**

328 The CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property is an array that  
329 contains the supported requested states for the instance of CIM\_Sensor or CIM\_NumericSensor. This  
330 property shall be the super set of the values to be used as the RequestedState parameter in the  
331 RequestStateChange() method.

332 The value of the CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall be  
333 an empty array or contain any combination of the following values: 2 (Enabled), 3 (Disabled), or  
334 11 (Reset).

### 335 **7.13.2 CIM\_EnabledLogicalElementCapabilities.ElementNameEditSupported**

336 The CIM\_EnabledLogicalElementCapabilities.ElementNameEditSupported property shall have a value of  
337 TRUE when the implementation supports client modification of the ElementName property of the  
338 associated CIM\_Sensor or CIM\_NumericSensor instance.

### 339 **7.13.3 CIM\_EnabledLogicalElement.MaxElementNameLen**

340 The CIM\_EnabledLogicalElementCapabilities.MaxElementNameLen property shall be implemented when  
341 the CIM\_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE.

## 342 **7.14 Sensor State Management**

343 Sensor state management is optional. Sensor state management consists of the RequestStateChange()  
344 method of the Central Instance being supported (see 8.1) and the RequestedState property of the Central  
345 Instance having a value other than 12 (Not Applicable).

### 346 **7.14.1 Sensor State Management Support**

347 When no CIM\_EnabledLogicalElementCapabilities instance is associated with the Central Instance,  
348 sensor state management shall not be supported.

349 When a CIM\_EnabledLogicalElementCapabilities instance is associated with the Central Instance but the  
350 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property is an empty array, sensor  
351 state management shall not be supported.

352 When a CIM\_EnabledLogicalElementCapabilities instance is associated with the Central Instance and the  
353 CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported property is a non-empty array,  
354 sensor state management shall be supported.

## 355 **7.15 CIM\_Sensor.RequestedState and CIM\_NumericSensor.RequestedState**

356 The RequestedState property shall have a value of 12 (Not Applicable), a value of 5 (No Change), or a  
357 value that is contained in the CIM\_EnabledLogicalElementCapabilities.RequestedStatesSupported  
358 property array of the associated CIM\_EnabledLogicalElementCapabilities instance (see 7.13.1).

359 When sensor state management is supported and the RequestStateChange() method is successfully  
360 executed, the RequestedState property shall be set to the value of the RequestedState parameter of the  
361 RequestStateChange() method. After the RequestStateChange() method completes successfully, the  
362 RequestedState and EnabledState properties shall have equal values with the exception of the  
363 transitional requested state 11 (Reset). The value of the RequestedState property may also change as a  
364 result of the request for a change to the sensor's enabled state by a non-CIM implementation.

### 365 **7.15.1 RequestedState — 12 (Not Applicable)**

366 When sensor state management is not supported, the value of the RequestedState property of the  
367 Central Instance shall be 12 (Not Applicable).

### 368 7.15.2 RequestedState — 5 (No Change)

369 When sensor state management is supported, the initial value of the RequestedState property of the  
370 Central Instance shall be 5 (No Change).

### 371 7.16 CIM\_Sensor.EnabledState and CIM\_NumericSensor.EnabledState

372 Table 4 describes the mapping between the EnabledState property values and the corresponding  
373 description of the state of the sensor. The EnabledState property shall match the values that are specified  
374 in Table 4. The value of the EnabledState property may change as a result of a client execution of the  
375 RequestStateChange() method or a change to the sensor's enabled state by a non-CIM implementation.

376

**Table 4 – EnabledState Value Description**

Value	Description	Extended Description
2	Enabled	Sensor shall be operational.
3	Disabled	Sensor shall be disabled.
5	Not Applicable	Sensor state is indeterminate, or sensor state management is not supported.

### 377 7.17 CIM\_Sensor.OtherSensorTypeDescription and 378 CIM\_NumericSensor.OtherSensorTypeDescription

379 The OtherSensorTypeDescription property shall be mandatory when the SensorType property is set to a  
380 value of 1 (Other).

381 The OtherSensorTypeDescription property shall be formatted as a free-formed string of variable length  
382 (pattern ".\*").

### 383 7.18 CIM\_SystemDevice and CIM\_AssociatedSensor

384 When the Central Instance represents a sensor for the entire managed system, the instance of  
385 CIM\_ComputerSystem that is referenced by CIM\_SystemDevice shall identify the managed system and  
386 no instances of CIM\_AssociatedSensor shall reference the Central Instance.

387 When the Central Instance represents a sensor for one or more discrete components and not for the  
388 entire system, for each component an instance of CIM\_AssociatedSensor shall reference the Central  
389 Instance and the CIM\_ManagedElement that identifies the component.

### 390 7.19 CIM\_Sensor.ElementName and CIM\_NumericSensor.ElementName

391 The ElementName property shall be formatted as a free-formed string of variable length (pattern ".\*").

#### 392 7.19.1 ElementName Is Modifiable

393 Implementations may allow the ElementName property to be modifiable by a client. This behavior is  
394 conditional. This section describes the CIM elements and behavior requirements when an implementation  
395 supports client modification of the ElementName property.

396 **8 Methods**

397 This section details the requirements for supporting intrinsic operations and extrinsic methods for the CIM  
 398 elements defined by this profile.

399 **8.1 CIM\_Sensor.RequestStateChange() and**  
 400 **CIM\_NumericSensor.RequestStateChange()**

401 The RequestStateChange() method shall be implemented when sensor state management is supported  
 402 (see 7.14.1).

403 Invocation of the RequestStateChange() method shall change the element's state to the value specified  
 404 in the RequestedState parameter.

405 RequestStateChange() return code values are specified in Table 5. RequestStateChange() parameters  
 406 are specified in Table 6.

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

409 No standard messages are defined for this method.

410 **Table 5 – RequestStateChange() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
1	Method is not supported in the implementation.
2	Error occurred.

411 **Table 6 – RequestStateChange() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	RequestedState	uint16	Shall have a value from the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported array (see 7.14)
OUT	Job	CIM_ConcreteJob REF	Returned if job started
IN, REQ	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

412 **8.2 CIM\_NumericSensor.RestoreDefaultThresholds()**

413 The CIM\_NumericSensor.RestoreDefaultThresholds() method shall be implemented and shall not return  
 414 a value of 1 (Unsupported) when the CIM\_NumericSensor.SettableThresholds property is a non-empty  
 415 array (see 7.12).

416 Invocation of the CIM\_NumericSensor.RestoreDefaultThresholds() method shall reset the values of the  
 417 thresholds of the sensor represented by the instance of CIM\_NumericSensor to the hardware defaults.

418 Detailed requirements of the CIM\_NumericSensor.RestoreDefaultThresholds() method are specified in  
 419 Table 7.

420 No method parameters are defined for this method.

421 No standard messages are defined for this method.

422 **Table 7 – CIM\_NumericSensor.RestoreDefaultThresholds() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
1	Method is not supported in the implementation.
2	Error occurred.

### 423 8.3 Profile Conventions for Operations

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

426 The default list of operations for all classes is as follows:

427     GetInstance( )

428     EnumerateInstances( )

429     EnumerateInstanceNames( )

430 For classes that are referenced by an association, the default list also includes the following operations:

431     Associators( )

432     AssociatorNames( )

433     References( )

434     ReferenceNames( )

### 435 8.4 CIM\_Sensor

436 Table 8 lists implementation requirements for operations. If implemented, these operations shall be  
437 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 8, all operations in  
438 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

440 **Table 8 – Operations: CIM\_Sensor**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.4.1.	None

#### 441 8.4.1 CIM\_Sensor — ModifyInstance

442 This section details the requirements for the ModifyInstance operation applied to an instance of  
443 CIM\_Sensor. The ModifyInstance operation may be supported.

444 The ModifyInstance operation shall be supported when the ElementNameEditSupported property of the  
445 CIM\_EnabledLogicalElementCapabilities instance that is associated with the CIM\_Sensor instance has a  
446 value of TRUE.



447 **8.4.1.1 CIM\_Sensor.ElementName**

448 When the ElementNameEditSupported property of the CIM\_EnabledLogicalElementCapabilities instance  
 449 that is associated with the CIM\_Sensor instance has a value of TRUE, the implementation shall allow the  
 450 ModifyInstance operation to change the value of the ElementName property of the CIM\_Sensor instance.  
 451 The ModifyInstance operation shall enforce the length restriction specified in the MaxElementNameLen  
 452 property of the CIM\_EnabledLogicalElementCapabilities instance.

453 When the ElementNameEditSupported property of the CIM\_EnabledLogicalElementCapabilities instance  
 454 has a value of FALSE, the implementation shall not allow the ModifyInstance operation to change the  
 455 value of the ElementName property of the CIM\_Sensor instance.

456 **8.5 CIM\_NumericSensor**

457 Table 9 lists implementation requirements for operations. If implemented, these operations shall be  
 458 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 9, all operations in  
 459 the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

461 **Table 9 – Operations: CIM\_NumericSensor**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.5.1.	None

462 **8.5.1 CIM\_NumericSensor — ModifyInstance**

463 This section details the requirements for the ModifyInstance operation applied to an instance of  
 464 CIM\_NumericSensor. The ModifyInstance operation may be supported.

465 The ModifyInstance operation shall be supported when the ElementNameEditSupported property of the  
 466 CIM\_EnabledLogicalElementCapabilities instance that is associated with the CIM\_NumericSensor  
 467 instance has a value of TRUE.

468 The ModifyInstance operation shall be supported when CIM\_NumericSensor.SettableThresholds property  
 469 is a non-empty array. The ModifyInstance operation shall modify the following properties:

- 470 • LowerThresholdNonCritical when CIM\_NumericSensor.SettableThresholds contains a value of 0  
 471 (LowerThresholdNonCritical)
- 472 • UpperThresholdNonCritical when CIM\_NumericSensor.SettableThresholds contains a value of 1  
 473 (UpperThresholdNonCritical)
- 474 • LowerThresholdCritical when CIM\_NumericSensor.SettableThresholds contains a value of 2  
 475 (LowerThresholdCritical)
- 476 • UpperThresholdCritical when CIM\_NumericSensor.SettableThresholds contains a value of 3  
 477 (UpperThresholdCritical)
- 478 • LowerThresholdFatal when CIM\_NumericSensor.SettableThresholds contains a value of 4  
 479 (LowerThresholdFatal)
- 480 • UpperThresholdFatal when CIM\_NumericSensor.SettableThresholds contains a value of 5  
 481 (UpperThresholdFatal)

### 482 8.5.1.1 CIM\_NumericSensor.ElementName

483 When the ElementNameEditSupported property of the CIM\_EnabledLogicalElementCapabilities instance  
 484 that is associated with the CIM\_NumericSensor instance has a value of TRUE, the implementation shall  
 485 allow the ModifyInstance operation to change the value of the ElementName property of the  
 486 CIM\_NumericSensor instance. The ModifyInstance operation shall enforce the length restriction specified  
 487 in the MaxElementNameLen property of the CIM\_EnabledLogicalElementCapabilities instance.

488 When the ElementNameEditSupported property of the CIM\_EnabledLogicalElementCapabilities instance  
 489 has a value of FALSE, the implementation shall not allow the ModifyInstance operation to change the  
 490 value of the ElementName property of the CIM\_NumericSensor instance.

## 491 8.6 CIM\_EnabledLogicalElementCapabilities

492 All operations in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

## 494 8.7 CIM\_ElementCapabilities

495 Table 10 lists implementation requirements for operations. If implemented, these operations shall be  
 496 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 10, all operations  
 497 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

499 **Table 10 – Operations: CIM\_ElementCapabilities**

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

## 500 8.8 CIM\_SystemDevice

501 Table 11 lists implementation requirements for operations. If implemented, these operations shall be  
 502 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 11, all operations  
 503 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

505 **Table 11 – Operations: CIM\_SystemDevice**

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

## 506 8.9 CIM\_AssociatedSensor

507 Table 12 lists implementation requirements for operations. If implemented, these operations shall be  
 508 implemented as defined in [DSP0200](#). In addition, and unless otherwise stated in Table 12, all operations  
 509 in the default list in 8.3 shall be implemented as defined in [DSP0200](#).

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

511 **Table 12 – Operations: CIM\_AssociatedSensor**

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

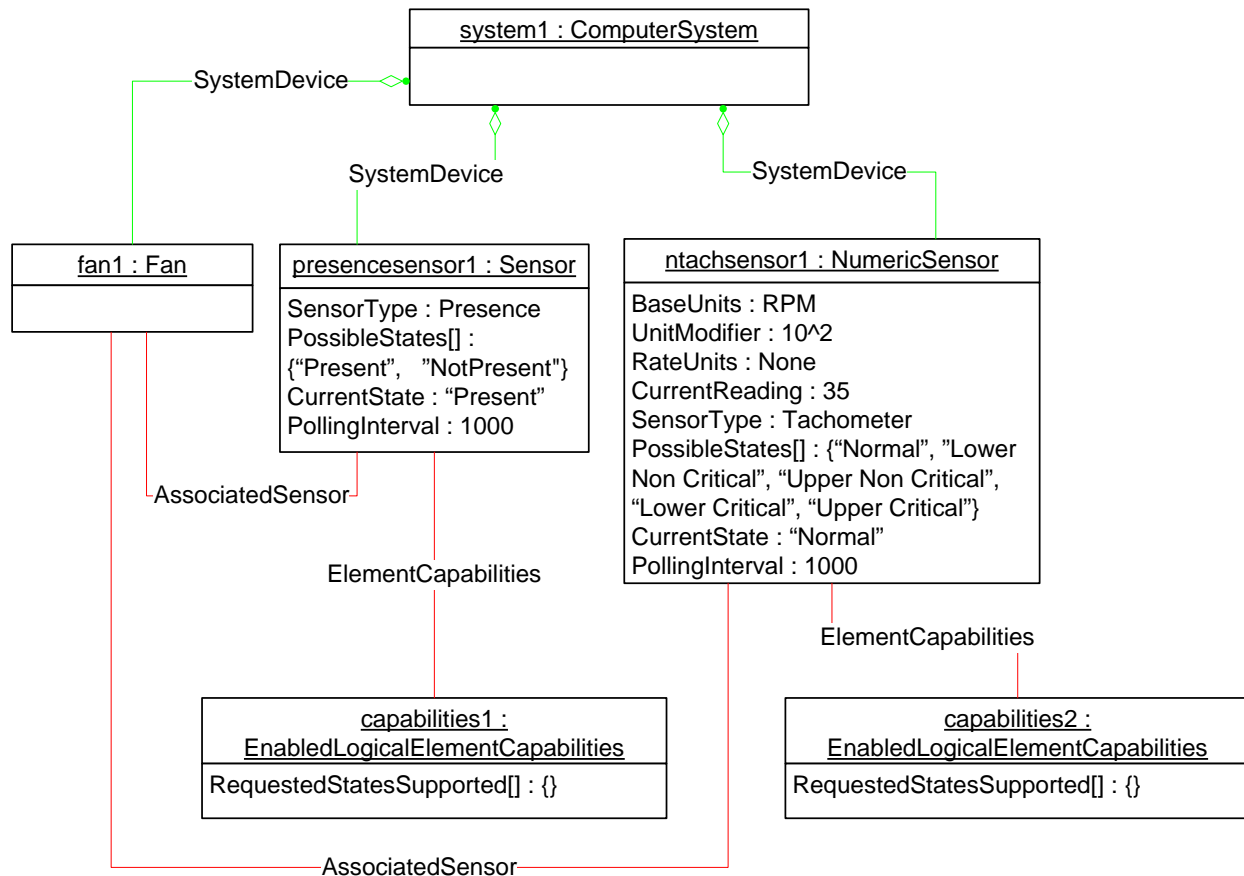
## 512 9 Use Cases

513 All use cases are based on the implementation conformance to the DMTF *Sensors Profile*.

### 514 9.1 Object Diagrams

515 Figure 2 represents a possible instantiation of the *Sensors Profile* classes. In the diagram, the instances  
 516 of CIM\_Sensor and CIM\_NumericSensor are associated with an instance of CIM\_Fan through instances  
 517 of CIM\_AssociatedSensor. In other cases, different instances of concrete classes derived from  
 518 CIM\_ManagedElement can be associated through CIM\_AssociatedSensor with an instance of  
 519 CIM\_Sensor or CIM\_NumericSensor.

520 Based on the SensorType property of the CIM\_Sensor instance, presencesensor1 is a Presence sensor.  
 521 presencesensor1 is a discrete sensor and provides the value “Present” or “Not Present” based on the  
 522 PossibleStates property. Based on the SensorType property, ntachsensorn1 is a Tachometer sensor,  
 523 which is an analog sensor, and provides numeric reading of the fan1 speed. Based on the BaseUnits  
 524 property, the reading is in RPM units, and the RateUnit property shows no additional units. The  
 525 CurrentReading property in this object diagram has a value of 35, which is multiplied by the value of the  
 526 UnitModifier property (in this case 10<sup>2</sup>), and therefore has a value of 3500 RPM.



527

528

**Figure 2 – Sensors Profile: Object Diagram**

## 529 9.2 Show All Current States of the Monitored Devices in the Computer System

530 Starting from the `CIM_ComputerSystem` instance that represents the computer system, select all of the  
 531 `CIM_Sensor` and `CIM_NumericSensor` instances that are associated through instances of  
 532 `CIM_SystemDevice`, which represent the sensors of the computer system. Iterate through the instances  
 533 and get the `CIM_Sensor.CurrentState` and `CIM_NumericSensor.CurrentState` property value, which will  
 534 represent the states of the monitored devices.

## 535 9.3 Find Sensors Associated with a Specific Component

536 Select all of the `CIM_Sensor` and `CIM_NumericSensor` instances that are associated with the instance of  
 537 a subclass of `CIM_ManagedElement` through an instance of `CIM_AssociatedSensor`.

## 538 9.4 Change Upper Non-Critical Threshold of Numeric Sensor

539 Select the instance of `CIM_NumericSensor`. Determine whether the `SettableThresholds` property contains  
 540 a value of 1 (`UpperThresholdNonCritical`). If so, set the value of the `UpperThresholdNonCritical` property;  
 541 otherwise, return an appropriate error.

## 542 9.5 Reset Sensor to Threshold Defaults

543 Select the instance of `CIM_NumericSensor`. Execute the method `RestoreDefaultThresholds()`.

544 **9.6 Determining If ElementName Can Be Modified**

545 For a given instance of CIM\_Sensor or CIM\_NumericSensor, a client can determine whether it can modify  
 546 the ElementName as follows:

- 547 1) Find the CIM\_EnabledLogicalElementCapabilities instance that is associated with the target  
 548 instance.
- 549 2) Query the value of the ElementNameEditSupported property of the  
 550 CIM\_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify  
 551 the ElementName property of the target instance.

552 **9.7 Determining If State Management Is Supported**

- 553 1) For a given instance of CIM\_Sensor or CIM\_NumericSensor, a client can determine whether  
 554 state management is supported as follows:
- 555 2) Find the CIM\_EnabledLogicalElementCapabilities instance that is associated with the instance.
- 556 3) Query the value of the RequestedStatesSupported property. If at least one value is specified,  
 557 state management is supported.

558 **10 CIM Elements**

559 Table 13 shows the list of CIM Elements for this profile and details their requirements. The  
 560 implementation requirements for the classes and properties described in this section are defined in 7,  
 561 "Implementation Requirements".

562 **Table 13 – CIM Elements: Sensors Profile**

Element Name	Requirement	Description
<b>Classes</b>		
CIM_Sensor	Conditional	See 7.1 and 10.1.
CIM_NumericSensor	Conditional	See 7.1 and 10.2.
CIM_EnabledLogicalElementCapabilities	Optional	See 7.13 and 10.3.
CIM_ElementCapabilities	Optional	See 10.4.
CIM_SystemDevice	Mandatory	See 7.18 and 10.5.
CIM_AssociatedSensor	Optional	See 7.18 and 10.6.
CIM_RegisteredProfile	Mandatory	See 10.7.
<b>Indications</b>		
None defined in this profile		

563 **10.1 CIM\_Sensor**

564 CIM\_Sensor is used to represent a discrete sensor. The CIM\_Sensor class is mandatory if the  
 565 CIM\_NumericSensor class is not implemented. Table 14 provides information about the properties of  
 566 CIM\_Sensor.

567

**Table 14 – Class: CIM\_Sensor**

Properties and Methods	Requirement	Notes
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
DeviceID	Mandatory	Key
SensorType	Mandatory	None
PossibleStates	Mandatory	See 7.2.
CurrentState	Mandatory	See 7.4.
ElementName	Mandatory	See 7.13.2.
OtherSensorTypeDescription	Conditional	See 7.17.
EnabledState	Mandatory	See 7.16.
RequestedState	Mandatory	See 7.14.
OperationalStatus	Mandatory	None
HealthState	Mandatory	None
RequestStateChange( )	Conditional	See 8.1.

568 **10.2 CIM\_NumericSensor**

569 CIM\_NumericSensor is used to represent an analog sensor. The CIM\_NumericSensor class is mandatory  
 570 when the CIM\_Sensor class is not implemented. Table 15 provides information about the properties of  
 571 CIM\_NumericSensor.

572

**Table 15 – Class: CIM\_NumericSensor**

Properties and Methods	Requirement	Notes
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
DeviceID	Mandatory	Key
BaseUnits	Mandatory	None
UnitModifier	Mandatory	None
RateUnits	Mandatory	None
CurrentReading	Mandatory	None
LowerThresholdNonCritical	Conditional	See 7.5.
UpperThresholdNonCritical	Conditional	See 7.6.
LowerThresholdCritical	Conditional	See 7.7.
UpperThresholdCritical	Conditional	See 7.8.
LowerThresholdFatal	Conditional	See 7.9.
UpperThresholdFatal	Conditional	See 7.10.
SupportedThresholds	Mandatory	See 7.11.
SettableThresholds	Mandatory	See 7.12.
SensorType	Mandatory	None
PossibleStates	Mandatory	See 7.3.
CurrentState	Mandatory	See 7.4.
ElementName	Mandatory	See 7.13.2.
OtherSensorTypeDescription	Conditional	See 7.17.
EnabledState	Mandatory	See 7.16.
RequestedState	Mandatory	See 7.14.
OperationalStatus	Mandatory	None
HealthState	Mandatory	None
RequestStateChange( )	Conditional	See 8.1.
RestoreDefaultThresholds( )	Conditional	See 8.2.

### 573 10.3 CIM\_EnabledLogicalElementCapabilities

574 CIM\_EnabledLogicalElementCapabilities is used to represent the capabilities of the sensor as it applies to  
 575 the properties of CIM\_Sensor or CIM\_NumericSensor that are derived from CIM\_EnabledLogicalElement,  
 576 such as RequestedEnabledState. For implementation details, see 7.13.

577 Table 16 provides information about the properties of CIM\_EnabledLogicalElementCapabilities.

578 **Table 16 – Class: CIM\_EnabledLogicalElementCapabilities**

Properties	Requirement	Notes
InstanceID	Mandatory	Key
RequestedStatesSupported	Mandatory	See 7.13.1.
ElementNameEditSupported	Mandatory	See 7.13.2.
MaxElementNameLen	Conditional	See 7.13.3.

### 579 10.4 CIM\_ElementCapabilities

580 CIM\_ElementCapabilities is used to associate CIM\_Sensor or CIM\_NumericSensor with an instance of  
 581 CIM\_EnabledLogicalElementCapabilities that describes the capabilities of CIM\_Sensor or  
 582 CIM\_NumericSensor. Table 17 provides information about the properties of CIM\_ElementCapabilities.

583 **Table 17 – Class: CIM\_ElementCapabilities**

Properties	Requirement	Notes
ManagedElement	Mandatory	Key
Capabilities	Mandatory	Key. See 7.13.

### 584 10.5 CIM\_SystemDevice

585 CIM\_SystemDevice is used to associate the instance of CIM\_Sensor or CIM\_NumericSensor with the  
 586 instance of CIM\_ComputerSystem of which the CIM\_Sensor instance is a member. Table 18 provides  
 587 information about the properties of CIM\_SystemDevice.

588 **Table 18 – Class: CIM\_SystemDevice**

Properties	Requirement	Notes
GroupComponent	Mandatory	<b>Key:</b> shall be a reference to the CIM_ComputerSystem instance of which the current CIM_Sensor or CIM_NumericSensor instance is a member.
PartComponent	Mandatory	<b>Key:</b> shall be a reference to the current CIM_Sensor or CIM_NumericSensor instance.



589 **10.6 CIM\_AssociatedSensor**

590 CIM\_AssociatedSensor is used to associate the instance of CIM\_Sensor or CIM\_NumericSensor with the  
 591 instance of a subclass of CIM\_ManagedElement. Table 19 provides information about the properties of  
 592 CIM\_AssociatedSensor.

593 **Table 19 – Class: CIM\_AssociatedSensor**

Properties	Requirement	Notes
Antecedent	Mandatory	Shall be a reference to a specific instance of CIM_Sensor or CIM_NumericSensor.
Dependent	Mandatory	Shall be a reference to the instance of a subclass of CIM_ManagedElement that the sensor is monitoring.

594 **10.7 CIM\_RegisteredProfile**

595 CIM\_RegisteredProfile is defined by the [Profile Registration Profile](#). The requirements denoted in  
 596 Table 20 are in addition to those mandated by the [Profile Registration Profile](#).

597 **Table 20 – Class: CIM\_RegisteredProfile**

Properties	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of "Sensors".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.2".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

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

601  
602  
603  
604

## ANNEX A (informative)

### Change Log

Version	Date	Description
1.0.0c	05/16/2006	Initial Preliminary Release
1.0.0	11/06/2007	Final Specification Release
1.0.1	09/25/2008	Errata 1.0.1
1.0.2	10/28/2009	Errata 1.0.2: Changed the values for the EnabledState property.

605