



1
2
3
4
5

Document Number: DSP1009

Date: 2008-09-29

Version: 1.0.1

6 **Sensors Profile**

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

10 Copyright Notice

11 Copyright © 2008 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27 withdrawn or modified after publication, and shall be indemnified and held harmless by any party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 References under Development.....	7
40	2.3 Other References.....	7
41	3 Terms and Definitions	7
42	4 Symbols and Abbreviated Terms	8
43	5 Synopsis.....	9
44	6 Description	9
45	7 Implementation Requirements	10
46	7.1 CIM_Sensor and CIM_NumericSensor	10
47	7.2 CIM_Sensor.PossibleStates	10
48	7.3 CIM_NumericSensor.PossibleStates.....	10
49	7.4 CIM_Sensor.CurrentState and CIM_NumericSensor.CurrentState	11
50	7.5 CIM_NumericSensor.LowerThresholdNonCritical.....	11
51	7.6 CIM_NumericSensor.UpperThresholdNonCritical.....	11
52	7.7 CIM_NumericSensor.LowerThresholdCritical	11
53	7.8 CIM_NumericSensor.UpperThresholdCritical	12
54	7.9 CIM_NumericSensor.LowerThresholdFatal	12
55	7.10 CIM_NumericSensor.UpperThresholdFatal	12
56	7.11 CIM_NumericSensor.SupportedThresholds.....	12
57	7.12 CIM_NumericSensor.SettableThresholds	12
58	7.13 CIM_EnabledLogicalElementCapabilities.....	12
59	7.14 Sensor State Management	13
60	7.15 CIM_Sensor.RequestedState and CIM_NumericSensor.RequestedState	13
61	7.16 CIM_Sensor.EnabledState and CIM_NumericSensor.EnabledState.....	14
62	7.17 CIM_Sensor.OtherSensorTypeDescription and	
63	CIM_NumericSensor.OtherSensorTypeDescription.....	14
64	7.18 CIM_SystemDevice and CIM_AssociatedSensor	14
65	7.19 CIM_Sensor.ElementName and CIM_NumericSensor.ElementName	14
66	8 Methods.....	15
67	8.1 CIM_Sensor.RequestStateChange() and CIM_NumericSensor.RequestStateChange()	15
68	8.2 CIM_NumericSensor.RestoreDefaultThresholds()	15
69	8.3 Profile Conventions for Operations.....	16
70	8.4 CIM_Sensor	16
71	8.5 CIM_NumericSensor.....	17
72	8.6 CIM_EnabledLogicalElementCapabilities.....	18
73	8.7 CIM_ElementCapabilities	18
74	8.8 CIM_SystemDevice	18
75	8.9 CIM_AssociatedSensor	19
76	9 Use Cases.....	19
77	9.1 Object Diagrams	19
78	9.2 Show All Current States of the Monitored Devices in the Computer System.....	20
79	9.3 Find Sensors Associated with a Specific Component	20

80	9.4	Change Upper Non-Critical Threshold of Numeric Sensor.....	20
81	9.5	Reset Sensor to Threshold Defaults.....	20
82	9.6	Determining If ElementName Can Be Modified.....	21
83	9.7	Determining If State Management Is Supported	21
84	10	CIM Elements.....	21
85	10.1	CIM_Sensor.....	22
86	10.2	CIM_NumericSensor.....	23
87	10.3	CIM_EnabledLogicalElementCapabilities.....	24
88	10.4	CIM_ElementCapabilities	24
89	10.5	CIM_SystemDevice	24
90	10.6	CIM_AssociatedSensor	25
91	10.7	CIM_RegisteredProfile.....	25
92		ANNEX A (informative) Change Log.....	26
93		ANNEX B (informative) Acknowledgments	27

94

95 Figures

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

98

99 Tables

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

120

121

Foreword

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

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

125

Introduction

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

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

135

Sensors Profile

136 1 Scope

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

140 2 Normative References

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

144 2.1 Approved References

145 DMTF [DSP0200](#), *CIM Operations over HTTP 1.2.0*

146 DMTF [DSP0004](#), *CIM Infrastructure Specification 2.3.0*

147 DMTF [DSP1000](#), *Management Profile Specification Template*

148 DMTF [DSP1001](#), *Management Profile Specification Usage Guide*

149 2.2 References under Development

150 DMTF [DSP1033](#), *Profile Registration Profile*

151 2.3 Other References

152 ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
153 <http://isotc.iso.org>

154 Unified Modeling Language (UML) from the Open Management Group (OMG), <http://www.uml.org>

155 3 Terms and Definitions

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

157 3.1

158 **can**

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

160 3.2

161 **cannot**

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

163 3.3

164 **conditional**

165 indicates requirements to be followed strictly to conform to the document when the specified conditions
166 are met

- 167 **3.4**
168 **mandatory**
169 indicates requirements to be followed strictly to conform to the document and from which no deviation is
170 permitted
- 171 **3.5**
172 **may**
173 indicates a course of action permissible within the limits of the document
- 174 **3.6**
175 **need not**
176 indicates a course of action permissible within the limits of the document
- 177 **3.7**
178 **optional**
179 indicates a course of action permissible within the limits of the document
- 180 **3.8**
181 **referencing profile**
182 indicates a profile that owns the definition of this class and can include a reference to this profile in its
183 "Related Profiles" table
- 184 **3.9**
185 **shall**
186 indicates requirements to be followed strictly to conform to the document and from which no deviation is
187 permitted
- 188 **3.10**
189 **shall not**
190 indicates requirements to be followed strictly to conform to the document and from which no deviation is
191 permitted
- 192 **3.11**
193 **should**
194 indicates that among several possibilities, one is recommended as particularly suitable, without
195 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required
- 196 **3.12**
197 **should not**
198 indicates that a certain possibility or course of action is deprecated but not prohibited
- 199 **4 Symbols and Abbreviated Terms**
200 None.

201 5 Synopsis

202 **Profile Name:** Sensors

203 **Version:** 1.0.1

204 **Organization:** DMTF

205 **CIM Schema Version:** 2.19.1

206 **Central Class:** CIM_Sensor

207 **Scoping Class:** CIM_ComputerSystem

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

210 Table 1 identifies profiles related to this profile.

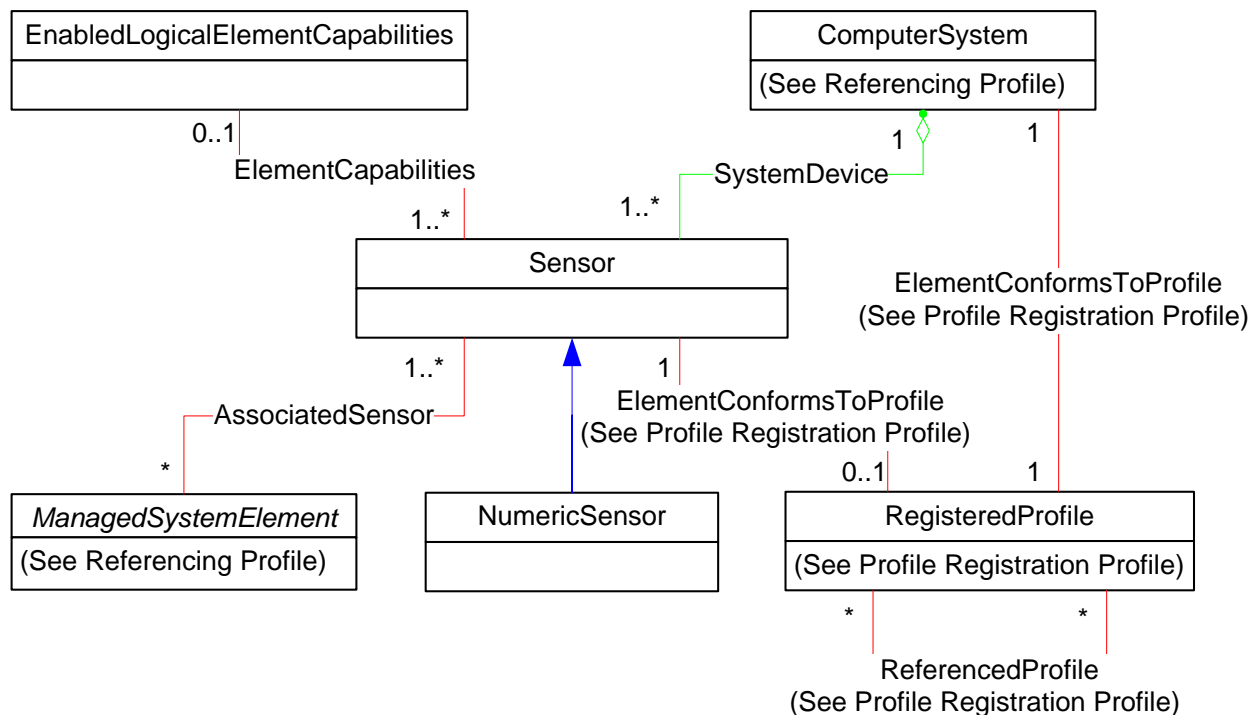
211 **Table 1 – Related Profiles**

Profile Name	Organization	Version	Requirement	Description
Profile Registration	DMTF	1.0.0	Mandatory	None

212 6 Description

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

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



216

217

Figure 1 – Sensors Profile: Class Diagram

218 7 Implementation Requirements

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

222 7.1 CIM_Sensor and CIM_NumericSensor

223 The implementation shall instantiate an instance of CIM_Sensor, including its subclass
 224 CIM_NumericSensor.

225 7.2 CIM_Sensor.PossibleStates

226 The CIM_Sensor.PossibleStates property shall represent an array of the possible string outputs of the
 227 sensor provided as a value of the CIM_Sensor.CurrentState property. The CIM_Sensor.SensorType
 228 property shall determine which CIM_Sensor.PossibleStates enumeration set to use.

229 The mappings between the CIM_Sensor.SensorType property values and the
 230 CIM_Sensor.PossibleStates property values are shown in Table 2. When the value of the
 231 CIM_Sensor.SensorType property matches a value in the "CIM_Sensor.SensorType" column of Table 2,
 232 the CIM_Sensor.PossibleStates property shall contain an array of values or an array of the subset of
 233 values specified in the corresponding "CIM_Sensor.PossibleStates" column. If the value of the
 234 CIM_Sensor.SensorType property is not listed in Table 2, the CIM_Sensor.PossibleStates property shall
 235 be defined by the implementation. The mapping between the values of CIM_Sensor.PossibleStates in
 236 Table 2 and the actual condition of the monitored device is implementation specific.

237 **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"

238 7.3 CIM_NumericSensor.PossibleStates

239 The CIM_NumericSensor.PossibleStates property shall represent an array of the possible string outputs
 240 of the sensor provided as a value of the CIM_NumericSensor.CurrentState property. The
 241 CIM_NumericSensor.SensorType property shall determine which CIM_NumericSensor.PossibleStates
 242 enumeration set to use.

243 The mappings between the CIM_NumericSensor.SensorType property values and the
 244 CIM_NumericSensor.PossibleStates property values are shown in Table 3. When the value of the
 245 CIM_NumericSensor.SensorType property matches a value in the “CIM_NumericSensor.SensorType”
 246 column of Table 3, the CIM_NumericSensor.PossibleStates property shall contain an array of values or
 247 an array of the subset of the values specified in the corresponding “CIM_NumericSensor.PossibleStates”
 248 column. If the value of the CIM_NumericSensor.SensorType property is not listed in Table 3, the
 249 CIM_NumericSensor.PossibleStates property shall be defined by the implementation. The mapping
 250 between the values of CIM_NumericSensor.PossibleStates in Table 3 and the actual condition of the
 251 monitored device is implementation specific.

252 **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)	

253 **7.4 CIM_Sensor.CurrentState and CIM_NumericSensor.CurrentState**

254 The CIM_Sensor.CurrentState property shall have a value of one of the elements in the
 255 CIM_Sensor.PossibleStates array.

256 The CIM_NumericSensor.CurrentState property shall have a value of one of the elements in the
 257 CIM_NumericSensor.PossibleStates array.

258 **7.5 CIM_NumericSensor.LowerThresholdNonCritical**

259 The CIM_NumericSensor.LowerThresholdNonCritical property shall be mandatory when the
 260 CIM_NumericSensor.SupportedThresholds array contains a value of 0 (LowerThresholdNonCritical).

261 The CIM_NumericSensor.LowerThresholdNonCritical property shall be settable only if the
 262 CIM_NumericSensor.SettableThresholds array contains a value of 0 (LowerThresholdNonCritical).

263 **7.6 CIM_NumericSensor.UpperThresholdNonCritical**

264 The CIM_NumericSensor.UpperThresholdNonCritical property shall be mandatory when the
 265 CIM_NumericSensor.SupportedThresholds array contains a value of 1 (UpperThresholdNonCritical).

266 The CIM_NumericSensor.UpperThresholdNonCritical property shall be settable only if the
 267 CIM_NumericSensor.SettableThresholds array contains a value of 1 (UpperThresholdNonCritical).

268 **7.7 CIM_NumericSensor.LowerThresholdCritical**

269 The CIM_NumericSensor.LowerThresholdCritical property shall be mandatory when the
 270 CIM_NumericSensor.SupportedThresholds array contains a value of 2 (LowerThresholdCritical).

271 The CIM_NumericSensor.LowerThresholdCritical property shall be settable only if the
272 CIM_NumericSensor.SettableThresholds array contains a value of 2 (LowerThresholdCritical).

273 **7.8 CIM_NumericSensor.UpperThresholdCritical**

274 The CIM_NumericSensor.UpperThresholdCritical property shall be mandatory when the
275 CIM_NumericSensor.SupportedThresholds array contains a value of 3 (UpperThresholdCritical).

276 The CIM_NumericSensor.UpperThresholdCritical property shall be settable only if the
277 CIM_NumericSensor.SettableThresholds array contains a value of 3 (UpperThresholdCritical).

278 **7.9 CIM_NumericSensor.LowerThresholdFatal**

279 The CIM_NumericSensor.LowerThresholdFatal property shall be mandatory when the
280 CIM_NumericSensor.SupportedThresholds array contains a value of 4 (LowerThresholdFatal).

281 The CIM_NumericSensor.LowerThresholdFatal property shall be settable only if the
282 CIM_NumericSensor.SettableThresholds array contains a value of 4 (LowerThresholdFatal).

283 **7.10 CIM_NumericSensor.UpperThresholdFatal**

284 The CIM_NumericSensor.UpperThresholdFatal property shall be mandatory when the
285 CIM_NumericSensor.SupportedThresholds array contains a value of 5 (UpperThresholdFatal).

286 The CIM_NumericSensor.UpperThresholdFatal property shall be settable only if the
287 CIM_NumericSensor.SettableThresholds array contains a value of 5 (UpperThresholdFatal).

288 **7.11 CIM_NumericSensor.SupportedThresholds**

289 The CIM_NumericSensor.SupportedThresholds property is an array that contains the list of the
290 implemented thresholds: LowerThresholdNonCritical, UpperThresholdNonCritical,
291 LowerThresholdCritical, UpperThresholdCritical, LowerThresholdFatal, and UpperThresholdFatal. When
292 the implementation does not support any of these threshold properties, the
293 CIM_NumericSensor.SupportedThresholds property shall be an empty array.

294 **7.12 CIM_NumericSensor.SettableThresholds**

295 The CIM_NumericSensor.SettableThresholds property is an array that contains the list of the settable
296 implemented thresholds: LowerThresholdNonCritical, UpperThresholdNonCritical,
297 LowerThresholdCritical, UpperThresholdCritical, LowerThresholdFatal, and UpperThresholdFatal. The
298 CIM_NumericSensor.SettableThresholds array shall contain the subset of values in the
299 CIM_NumericSensor.SupportedThresholds array (see 7.11). When the implementation does not support
300 any of the settable threshold properties, the CIM_NumericSensor.SettableThresholds property shall be an
301 empty array.

302 **7.13 CIM_EnabledLogicalElementCapabilities**

303 When the CIM_EnabledLogicalElementCapabilities class is instantiated, the instance of
304 CIM_EnabledLogicalElementCapabilities shall be associated with the Central Instance through the
305 CIM_ElementCapabilities association and used for advertising the capabilities of the Central Instance.

306 At most only one instance of CIM_EnabledLogicalElementCapabilities shall be associated with a given
307 instance of CIM_Sensor or CIM_NumericSensor.

308 **7.13.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported**

309 The CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property is an array that
310 contains the supported requested states for the instance of CIM_Sensor or CIM_NumericSensor. This
311 property shall be the super set of the values to be used as the RequestedState parameter in the
312 RequestStateChange() method.

313 The value of the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property shall be
314 an empty array or contain any combination of the following values: 2 (Enabled), 3 (Disabled), or
315 11 (Reset).

316 **7.13.2 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported**

317 The CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property shall have a value of
318 TRUE when the implementation supports client modification of the ElementName property of the
319 associated CIM_Sensor or CIM_NumericSensor instance.

320 **7.13.3 CIM_EnabledLogicalElement.MaxElementNameLen**

321 The CIM_EnabledLogicalElementCapabilities.MaxElementNameLen property shall be implemented when
322 the CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported property has a value of TRUE.

323 **7.14 Sensor State Management**

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

327 **7.14.1 Sensor State Management Support**

328 When no CIM_EnabledLogicalElementCapabilities instance is associated with the Central Instance,
329 sensor state management shall not be supported.

330 When a CIM_EnabledLogicalElementCapabilities instance is associated with the Central Instance but the
331 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property is an empty array, sensor
332 state management shall not be supported.

333 When a CIM_EnabledLogicalElementCapabilities instance is associated with the Central Instance and the
334 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported property is a non-empty array,
335 sensor state management shall be supported.

336 **7.15 CIM_Sensor.RequestedState and CIM_NumericSensor.RequestedState**

337 The RequestedState property shall have a value of 12 (Not Applicable), a value of 5 (No Change), or a
338 value that is contained in the CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported
339 property array of the associated CIM_EnabledLogicalElementCapabilities instance (see 7.13.1).

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

346 **7.15.1 RequestedState – 12 (Not Applicable)**

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

349 7.15.2 RequestedState – 5 (No Change)

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

352 7.16 CIM_Sensor.EnabledState and CIM_NumericSensor.EnabledState

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

357 **Table 4 – EnabledState Value Description**

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

358 7.17 CIM_Sensor.OtherSensorTypeDescription and 359 CIM_NumericSensor.OtherSensorTypeDescription

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

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

364 7.18 CIM_SystemDevice and CIM_AssociatedSensor

365 When the Central Instance represents a sensor for the entire managed system, the instance of
366 CIM_ComputerSystem that is referenced by CIM_SystemDevice shall identify the managed system and
367 no instances of CIM_AssociatedSensor shall reference the Central Instance.

368 When the Central Instance represents a sensor for one or more discrete components and not for the
369 entire system, for each component an instance of CIM_AssociatedSensor shall reference the Central
370 Instance and the CIM_ManagedElement that identifies the component.

371 7.19 CIM_Sensor.ElementName and CIM_NumericSensor.ElementName

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

373 7.19.1 ElementName Is Modifiable

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

377 **8 Methods**

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

380 **8.1 CIM_Sensor.RequestStateChange() and**
381 **CIM_NumericSensor.RequestStateChange()**

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

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

386 RequestStateChange() return code values are specified in Table 5. RequestStateChange() parameters
387 are specified in Table 6.

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

390 No standard messages are defined for this method.

391 **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.

392 **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

393 **8.2 CIM_NumericSensor.RestoreDefaultThresholds()**

394 The CIM_NumericSensor.RestoreDefaultThresholds() method shall be implemented and shall not return
395 a value of 1 (Unsupported) when the CIM_NumericSensor.SettableThresholds property is a non-empty
396 array (see 7.12).

397 Invocation of the CIM_NumericSensor.RestoreDefaultThresholds() method shall reset the values of the
398 thresholds of the sensor represented by the instance of CIM_NumericSensor to the hardware defaults.

399 Detailed requirements of the CIM_NumericSensor.RestoreDefaultThresholds() method are specified in
400 Table 7.

401 No method parameters are defined for this method.

402 No standard messages are defined for this method.

403 **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.

404 8.3 Profile Conventions for Operations

405 Support for operations for each profile class (including associations) is specified in the following
 406 subclauses. Each subclause includes either the statement “All operations in the default list in 8.3 are
 407 supported as described by DSP0200 version 1.2” or a table listing all of the operations that are not
 408 supported by this profile or where the profile requires behavior other than that described by DSP0200.

409 The default list of operations is as follows:

- 410 • GetInstance
- 411 • EnumerateInstances
- 412 • EnumerateInstanceNames
- 413 • Associators
- 414 • AssociatorNames
- 415 • References
- 416 • ReferenceNames

417 A compliant implementation shall support all of the operations in the default list for each class, unless the
 418 “Requirement” column states something other than *Mandatory*.

419 8.4 CIM_Sensor

420 Table 8 lists operations that either have special requirements beyond those from DSP0200 or shall not be
 421 supported.

422 **Table 8 – Operations: CIM_Sensor**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.4.1.	None

423 8.4.1 CIM_Sensor—ModifyInstance

424 This section details the requirements for the ModifyInstance operation applied to an instance of
 425 CIM_Sensor. The ModifyInstance operation may be supported.

426 The ModifyInstance operation shall be supported when the ElementNameEditSupported property of the
 427 CIM_EnabledLogicalElementCapabilities instance that is associated with the CIM_Sensor instance has a
 428 value of TRUE.

429 8.4.1.1 CIM_Sensor.ElementName

430 When the ElementNameEditSupported property of the CIM_EnabledLogicalElementCapabilities instance
 431 that is associated with the CIM_Sensor instance has a value of TRUE, the implementation shall allow the

432 ModifyInstance operation to change the value of the ElementName property of the CIM_Sensor instance.
 433 The ModifyInstance operation shall enforce the length restriction specified in the MaxElementNameLen
 434 property of the CIM_EnabledLogicalElementCapabilities instance.

435 When the ElementNameEditSupported property of the CIM_EnabledLogicalElementCapabilities instance
 436 has a value of FALSE, the implementation shall not allow the ModifyInstance operation to change the
 437 value of the ElementName property of the CIM_Sensor instance.

438 **8.5 CIM_NumericSensor**

439 Table 9 lists operations that either have special requirements beyond those from DSP0200 or shall not be
 440 supported.

441 **Table 9 – Operations: CIM_NumericSensor**

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.5.1.	None

442 **8.5.1 CIM_NumericSensor—ModifyInstance**

443 This section details the requirements for the ModifyInstance operation applied to an instance of
 444 CIM_NumericSensor. The ModifyInstance operation may be supported.

445 The ModifyInstance operation shall be supported when the ElementNameEditSupported property of the
 446 CIM_EnabledLogicalElementCapabilities instance that is associated with the CIM_NumericSensor
 447 instance has a value of TRUE.

448 The ModifyInstance operation shall be supported when CIM_NumericSensor.SettableThresholds property
 449 is a non-empty array. The ModifyInstance operation shall modify the following properties:

- 450 • LowerThresholdNonCritical when CIM_NumericSensor.SettableThresholds contains a value of 0
 451 (LowerThresholdNonCritical)
- 452 • UpperThresholdNonCritical when CIM_NumericSensor.SettableThresholds contains a value of 1
 453 (UpperThresholdNonCritical)
- 454 • LowerThresholdCritical when CIM_NumericSensor.SettableThresholds contains a value of 2
 455 (LowerThresholdCritical)
- 456 • UpperThresholdCritical when CIM_NumericSensor.SettableThresholds contains a value of 3
 457 (UpperThresholdCritical)
- 458 • LowerThresholdFatal when CIM_NumericSensor.SettableThresholds contains a value of 4
 459 (LowerThresholdFatal)
- 460 • UpperThresholdFatal when CIM_NumericSensor.SettableThresholds contains a value of 5
 461 (UpperThresholdFatal)

462 **8.5.1.1 CIM_NumericSensor.ElementName**

463 When the ElementNameEditSupported property of the CIM_EnabledLogicalElementCapabilities instance
 464 that is associated with the CIM_NumericSensor instance has a value of TRUE, the implementation shall
 465 allow the ModifyInstance operation to change the value of the ElementName property of the
 466 CIM_NumericSensor instance. The ModifyInstance operation shall enforce the length restriction specified
 467 in the MaxElementNameLen property of the CIM_EnabledLogicalElementCapabilities instance.

468 When the ElementNameEditSupported property of the CIM_EnabledLogicalElementCapabilities instance
 469 has a value of FALSE, the implementation shall not allow the ModifyInstance operation to change the
 470 value of the ElementName property of the CIM_NumericSensor instance.

471 **8.6 CIM_EnabledLogicalElementCapabilities**

472 All operations in the default list in 8.3 are supported as described by DSP0200 version 1.2.

473 **8.7 CIM_ElementCapabilities**

474 Table 10 lists operations that either have special requirements beyond those from DSP0200 or shall not
 475 be supported.

476 **Table 10 – Operations: CIM_ElementCapabilities**

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

477 **8.8 CIM_SystemDevice**

478 Table 11 lists operations that either have special requirements beyond those from DSP0200 or shall not
 479 be supported.

480 **Table 11 – Operations: CIM_SystemDevice**

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

481 **8.9 CIM_AssociatedSensor**

482 Table 12 lists operations that either have special requirements beyond those from DSP0200 or shall not
 483 be supported.

484 **Table 12 – Operations: CIM_AssociatedSensor**

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

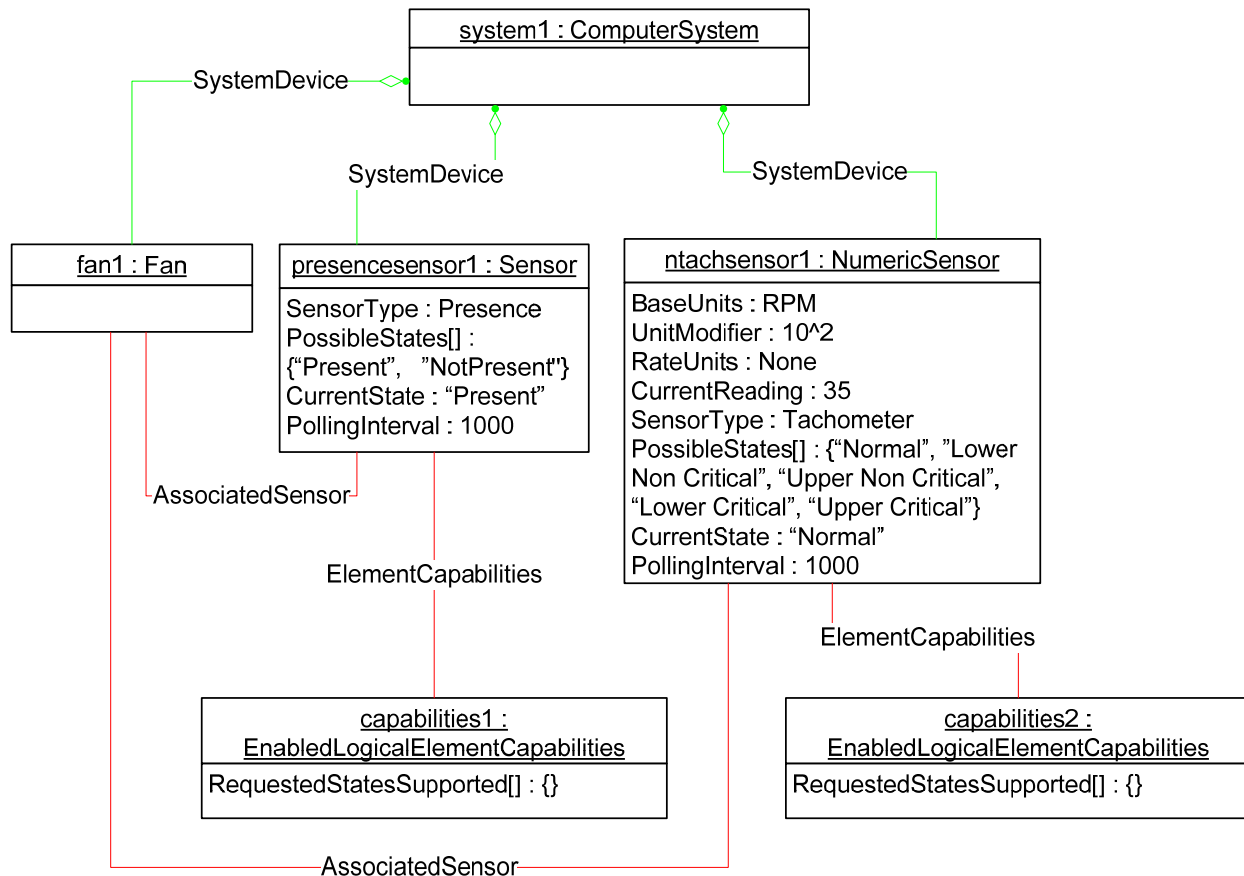
485 **9 Use Cases**

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

487 **9.1 Object Diagrams**

488 Figure 2 represents a possible instantiation of the *Sensors Profile* classes. In the diagram, the instances
 489 of CIM_Sensor and CIM_NumericSensor are associated with an instance of CIM_Fan through instances
 490 of CIM_AssociatedSensor. In other cases, different instances of concrete classes derived from
 491 CIM_ManagedElement can be associated through CIM_AssociatedSensor with an instance of
 492 CIM_Sensor or CIM_NumericSensor.

493 Based on the SensorType property of the CIM_Sensor instance, presencesensor1 is a Presence sensor.
 494 presencesensor1 is a discrete sensor and provides the value “Present” or “Not Present” based on the
 495 PossibleStates property. Based on the SensorType property, ntachsensor1 is a Tachometer sensor,
 496 which is an analog sensor, and provides numeric reading of the fan1 speed. Based on the BaseUnits
 497 property, the reading is in RPM units, and the RateUnit property shows no additional units. The
 498 CurrentReading property in this object diagram has a value of 35, which is multiplied by the value of the
 499 UnitModifier property (in this case 10²), and therefore has a value of 3500 RPM.



500
501

502

Figure 2 – Sensors Profile: Object Diagram

503 9.2 Show All Current States of the Monitored Devices in the Computer System

504 Starting from the CIM_ComputerSystem instance that represents the computer system, select all of the
505 CIM_Sensor and CIM_NumericSensor instances that are associated through instances of
506 CIM_SystemDevice, which represent the sensors of the computer system. Iterate through the instances
507 and get the CIM_Sensor.CurrentState and CIM_NumericSensor.CurrentState property value, which will
508 represent the states of the monitored devices.

509 9.3 Find Sensors Associated with a Specific Component

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

512 9.4 Change Upper Non-Critical Threshold of Numeric Sensor

513 Select the instance of CIM_NumericSensor. Determine whether the SettableThresholds property contains
514 a value of 1 (UpperThresholdNonCritical). If so, set the value of the UpperThresholdNonCritical property;
515 otherwise, return an appropriate error.

516 9.5 Reset Sensor to Threshold Defaults

517 Select the instance of CIM_NumericSensor. Execute the method RestoreDefaultThresholds().

518 **9.6 Determining If ElementName Can Be Modified**

519 For a given instance of CIM_Sensor or CIM_NumericSensor, a client can determine whether it can modify
 520 the ElementName as follows:

- 521 1) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the target
 522 instance.
- 523 2) Query the value of the ElementNameEditSupported property of the
 524 CIM_EnabledLogicalElementCapabilities instance. If the value is TRUE, the client can modify
 525 the ElementName property of the target instance.

526 **9.7 Determining If State Management Is Supported**

- 527 1) For a given instance of CIM_Sensor or CIM_NumericSensor, a client can determine whether
 528 state management is supported as follows:
- 529 2) Find the CIM_EnabledLogicalElementCapabilities instance that is associated with the instance.
- 530 3) Query the value of the RequestedStatesSupported property. If at least one value is specified,
 531 state management is supported.

532 **10 CIM Elements**

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

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

Element Name	Requirement	Description
Classes		
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.
Indications		
None defined in this profile		

537 **10.1 CIM_Sensor**

538 CIM_Sensor is used to represent a discrete sensor. The CIM_Sensor class is mandatory if the
 539 CIM_NumericSensor class is not implemented. Table 14 provides information about the properties of
 540 CIM_Sensor.

541

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.

542 **10.2 CIM_NumericSensor**

543 CIM_NumericSensor is used to represent an analog sensor. The CIM_NumericSensor class is mandatory
 544 when the CIM_Sensor class is not implemented. Table 15 provides information about the properties of
 545 CIM_NumericSensor.

546 **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.

547 10.3 CIM_EnabledLogicalElementCapabilities

548 CIM_EnabledLogicalElementCapabilities is used to represent the capabilities of the sensor as it applies to
 549 the properties of CIM_Sensor or CIM_NumericSensor that are derived from CIM_EnabledLogicalElement,
 550 such as RequestedEnabledState. For implementation details, see 7.13.

551 Table 16 provides information about the properties of CIM_EnabledLogicalElementCapabilities.

552 **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.

553 10.4 CIM_ElementCapabilities

554 CIM_ElementCapabilities is used to associate CIM_Sensor or CIM_NumericSensor with an instance of
 555 CIM_EnabledLogicalElementCapabilities that describes the capabilities of CIM_Sensor or
 556 CIM_NumericSensor. Table 17 provides information about the properties of CIM_ElementCapabilities.

557 **Table 17 – Class: CIM_ElementCapabilities**

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

558 10.5 CIM_SystemDevice

559 CIM_SystemDevice is used to associate the instance of CIM_Sensor or CIM_NumericSensor with the
 560 instance of CIM_ComputerSystem of which the CIM_Sensor instance is a member. Table 18 provides
 561 information about the properties of CIM_SystemDevice.

562 **Table 18 – Class: CIM_SystemDevice**

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

563 **10.6 CIM_AssociatedSensor**

564 CIM_AssociatedSensor is used to associate the instance of CIM_Sensor or CIM_NumericSensor with the
 565 instance of a subclass of CIM_ManagedElement. Table 19 provides information about the properties of
 566 CIM_AssociatedSensor.

567 **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.

568 **10.7 CIM_RegisteredProfile**

569 CIM_RegisteredProfile is defined by the Profile Registration Profile. The requirements denoted in
 570 Table 20 are in addition to those mandated by the Profile Registration Profile.

571 **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 "1.0.1".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

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

575
576
577
578

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

579

580
581
582
583

ANNEX B (informative)

Acknowledgments

584 The authors wish to acknowledge the following people.

585 **Editors:**

- 586 • Jon Hass – Dell Inc.
- 587 • Khachatur Papanyan – Dell Inc.
- 588 • Jim Davis (Errata) – WBEM Solutions

589 **Contributors:**

- 590 • Jon Hass – Dell Inc.
- 591 • Khachatur Papanyan – Dell Inc.
- 592 • Linda Martinez – Dell Inc.
- 593 • Jeff Hilland – HP
- 594 • Christina Shaw – HP
- 595 • Aaron Merkin – IBM
- 596 • Jeff Lynch – IBM
- 597 • Perry Vincent – Intel
- 598 • John Leung – Intel
- 599