Document Number: DSP1009

Date: 2012-08-07

Version: 1.1.0

Sensors Profile

Document Type: Specification

Document Status: DMTF Standard

Document Language: en-US

CONTENTS

Figures

Tables

Foreword

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

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

Acknowledgments

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

Editors:

- Jim Davis (Errata) – WBEM Solutions
- Jon Hass – Dell Inc.
- Steve Lee - Microsoft Corporation
- Khachatur Papanyan – Dell Inc.
- Hemal Shah – Broadcom Corporation

Contributors:

- Jim Davis – WBEM Solutions
- Jon Hass – Dell Inc.
- Jeff Hilland – Hewlett-Packard Company
- John Leung – Intel
- Jeff Lynch – IBM
- Linda Martinez – Dell Inc.
- Aaron Merkin – IBM
- Khachatur Papanyan – Dell Inc.
- Hemal Shah – Broadcom Corporation
- Christina Shaw – Hewlett-Packard Company
- Perry Vincent – Intel

Introduction

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

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

Sensors Profile

1 Scope

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

2 Normative references

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

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

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

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

DMTF DSP1000, *Management Profile Specification Template 1.0*
http://www.dmtf.org/sites/default/files/standards/documents/DSP1000.pdf

DMTF DSP1001, *Management Profile Specification Usage Guide 1.1,*
http://www.dmtf.org/standards/published_documents/DSP1001_1.1.pdf

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

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

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

3 Terms and definitions

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

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

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

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

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

3.1
conditional
indicates requirements to be followed strictly to conform to the document when the specified conditions are met

3.2
mandatory
indicates requirements to be followed strictly to conform to the document and from which no deviation is permitted

3.3
optional
indicates a course of action permissible within the limits of the document

3.4
referencing profile
indicates a profile that owns the definition of this class and can include a reference to this profile in its "Related Profiles" table

4 Symbols and abbreviated terms

The abbreviations defined in DSP0004, DSP0223, and DSP1001 apply to this document.

5 Synopsis

Profile name: Sensors

Version: 1.1.0

Organization: DMTF

CIM Schema version: 2.32

Central class: CIM_Sensor

Scoping class: CIM_ComputerSystem

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

Table 1 identifies profiles related to this profile.

Table 1 – Related profiles

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

6 Description

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

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

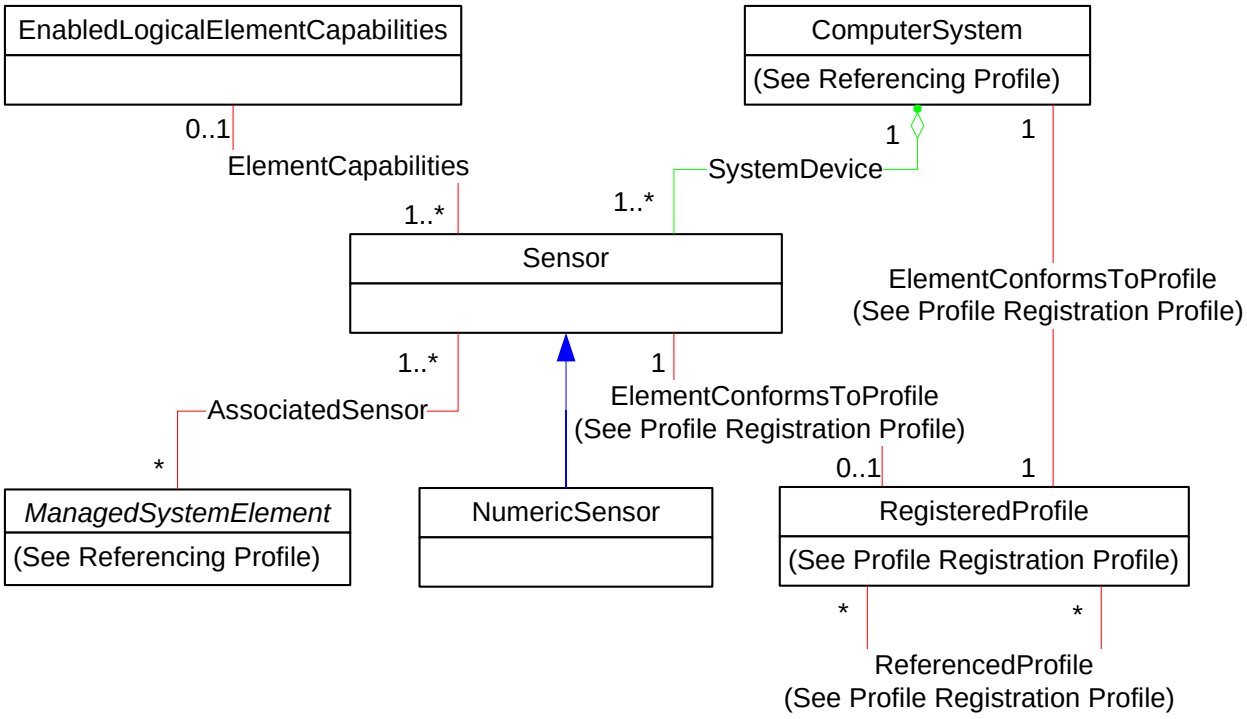

Figure 1 – Sensors Profile: Class diagram

7 Implementation requirements

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

7.1 CIM_Sensor and CIM_NumericSensor

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

7.2 CIM_Sensor.PossibleStates

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

240 The mappings between the CIM_Sensor.SensorType property values and the
241 CIM_Sensor.PossibleStates property values are shown in Table 2. When the value of the
242 CIM_Sensor.SensorType property matches a value in the “CIM_Sensor.SensorType” column of Table 2,
243 the CIM_Sensor.PossibleStates property shall contain an array of values or an array of the subset of
244 values specified in the corresponding “CIM_Sensor.PossibleStates” column. If the value of the
245 CIM_Sensor.SensorType property is not listed in Table 2, the CIM_Sensor.PossibleStates property shall
246 be defined by the implementation. The mapping between the values of CIM_Sensor.PossibleStates in
247 Table 2 and the actual condition of the monitored device is implementation specific.

248 **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”

249 ## 7.3 CIM_NumericSensor.PossibleStates

250 The CIM_NumericSensor.PossibleStates property shall represent an array of the possible string outputs
251 of the sensor provided as a value of the CIM_NumericSensor.CurrentState property. The
252 CIM_NumericSensor.SensorType property shall determine which CIM_NumericSensor.PossibleStates
253 enumeration set to use.

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

Table 3 – CIM_NumericSensor.PossibleStates values for numeric sensors

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

7.4 CIM_Sensor.CurrentState and CIM_NumericSensor.CurrentState

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

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

7.5 CIM_NumericSensor.LowerThresholdNonCritical

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

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

7.6 CIM_NumericSensor.UpperThresholdNonCritical

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

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

7.7 CIM_NumericSensor.LowerThresholdCritical

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

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

7.8 CIM_NumericSensor.UpperThresholdCritical

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

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

7.9 CIM_NumericSensor.LowerThresholdFatal

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

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

7.10 CIM_NumericSensor.UpperThresholdFatal

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

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

7.11 CIM_NumericSensor.SupportedThresholds

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

7.12 CIM_NumericSensor.SettableThresholds

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

7.13 CIM_EnabledLogicalElementCapabilities

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

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

7.13.1 CIM_EnabledLogicalElementCapabilities.RequestedStatesSupported

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

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

7.13.2 CIM_EnabledLogicalElementCapabilities.ElementNameEditSupported

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

7.13.3 CIM_EnabledLogicalElement.MaxElementNameLen

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

7.14 Sensor state management

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

7.14.1 Sensor state management support

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

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

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

7.15 CIM_Sensor.RequestedState and CIM_NumericSensor.RequestedState

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

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

7.15.1 RequestedState – 12 (Not Applicable)

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

7.15.2 RequestedState – 5 (No Change)

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

7.16 CIM_Sensor.EnabledState and CIM_NumericSensor.EnabledState

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

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.

7.17 CIM_Sensor.OtherSensorTypeDescription and CIM_NumericSensor.OtherSensorTypeDescription

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

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

7.18 CIM_SystemDevice and CIM_AssociatedSensor

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

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

7.19 CIM_Sensor.ElementName and CIM_NumericSensor.ElementName

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

7.19.1 ElementName is modifiable

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

7.20 CIM_Sensor.SensorContext

The SensorContext property indicates the purpose and context of the sensor. The property may indicate what entity is being monitored or where the sensor is installed.

The property value shall be formatted using the following algorithm: <OrgID>:<LocalID> where <OrgID> and <LocalID> are separated by a colon (:). <OrgID> shall include a unique name that is owned by the business entity creating or defining the SensorContext and <OrgID> shall not contain a colon (:). <LocalID> is chosen by the business entity and should not be reused to identify different underlying (real-world) elements.

For DMTF defined identifiers, the <OrgID> shall match "DMTF". Table 5 shows the values of SensorContext defined in this profile.

Table 5 – Sensor Context Values

SensorContext	Description
DMTF:Inlet	Sensor located at inlet
DMTF:Exhaust	Sensor located at exhaust
DMTF:CPU	Sensor monitoring a CPU
DMTF:Base Board	Sensor monitoring base board

8 Methods

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

8.1 CIM_Sensor.RequestStateChange( ) and CIM_NumericSensor.RequestStateChange( )

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

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

RequestStateChange( ) return code values are specified in Table 6. RequestStateChange( ) parameters are specified in Table 7.

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

No standard messages are defined for this method.

Table 6 – RequestStateChange( ) method: Return code values

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

414 **Table 7 – 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

415 ## 8.2 CIM_NumericSensor.RestoreDefaultThresholds( )

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

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

421 Detailed requirements of the CIM_NumericSensor.RestoreDefaultThresholds( ) method are specified in
422 Table 8.

423 No method parameters are defined for this method.

424 No standard messages are defined for this method.

425 **Table 8 – 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.

426 ## 8.3 Profile conventions for operations

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

431 The default list of operations is as follows:

432 • GetInstance
433 • EnumerateInstances
434 • EnumerateInstanceNames
435 • Associators
436 • AssociatorNames
437 • References
438 • ReferenceNames

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

8.4 CIM_Sensor

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

Table 9 – Operations: CIM_Sensor

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.4.1.	None

8.4.1 CIM_Sensor—ModifyInstance

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

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

8.4.1.1 CIM_Sensor.ElementName

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

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

8.5 CIM_NumericSensor

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

Table 10 – Operations: CIM_NumericSensor

Operation	Requirement	Messages
ModifyInstance	Optional. See 8.5.1.	None

8.5.1 CIM_NumericSensor—ModifyInstance

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

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

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

- LowerThresholdNonCritical when CIM_NumericSensor.SettableThresholds contains a value of 0 (LowerThresholdNonCritical)
- UpperThresholdNonCritical when CIM_NumericSensor.SettableThresholds contains a value of 1 (UpperThresholdNonCritical)
- LowerThresholdCritical when CIM_NumericSensor.SettableThresholds contains a value of 2 (LowerThresholdCritical)
- UpperThresholdCritical when CIM_NumericSensor.SettableThresholds contains a value of 3 (UpperThresholdCritical)
- LowerThresholdFatal when CIM_NumericSensor.SettableThresholds contains a value of 4 (LowerThresholdFatal)
- UpperThresholdFatal when CIM_NumericSensor.SettableThresholds contains a value of 5 (UpperThresholdFatal)

8.5.1.1 CIM_NumericSensor.ElementName

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

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

8.6 CIM_EnabledLogicalElementCapabilities

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

8.7 CIM_ElementCapabilities

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

Table 11 – Operations: CIM_ElementCapabilities

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

8.8 CIM_SystemDevice

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

Table 12 – Operations: CIM_SystemDevice

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

8.9 CIM_AssociatedSensor

Table 13 lists operations that either have special requirements beyond those from DSP0200 or shall not be supported.

Table 13 – Operations: CIM_AssociatedSensor

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

9 Use cases

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

9.1 Object diagrams

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

Based on the SensorType property of the CIM_Sensor instance, presencesensor1 is a Presence sensor. presencesensor1 is a discrete sensor and provides the value “Present” or “Not Present” based on the PossibleStates property. Based on the SensorType property, ntachsensor1 is a Tachometer sensor, which is an analog sensor, and provides numeric reading of the fan1 speed. Based on the BaseUnits property, the reading is in RPM units, and the RateUnit property shows no additional units. The CurrentReading property in this object diagram has a value of 35, which is multiplied by the value of the UnitModifier property (in this case 10^2), and therefore has a value of 3500 RPM.

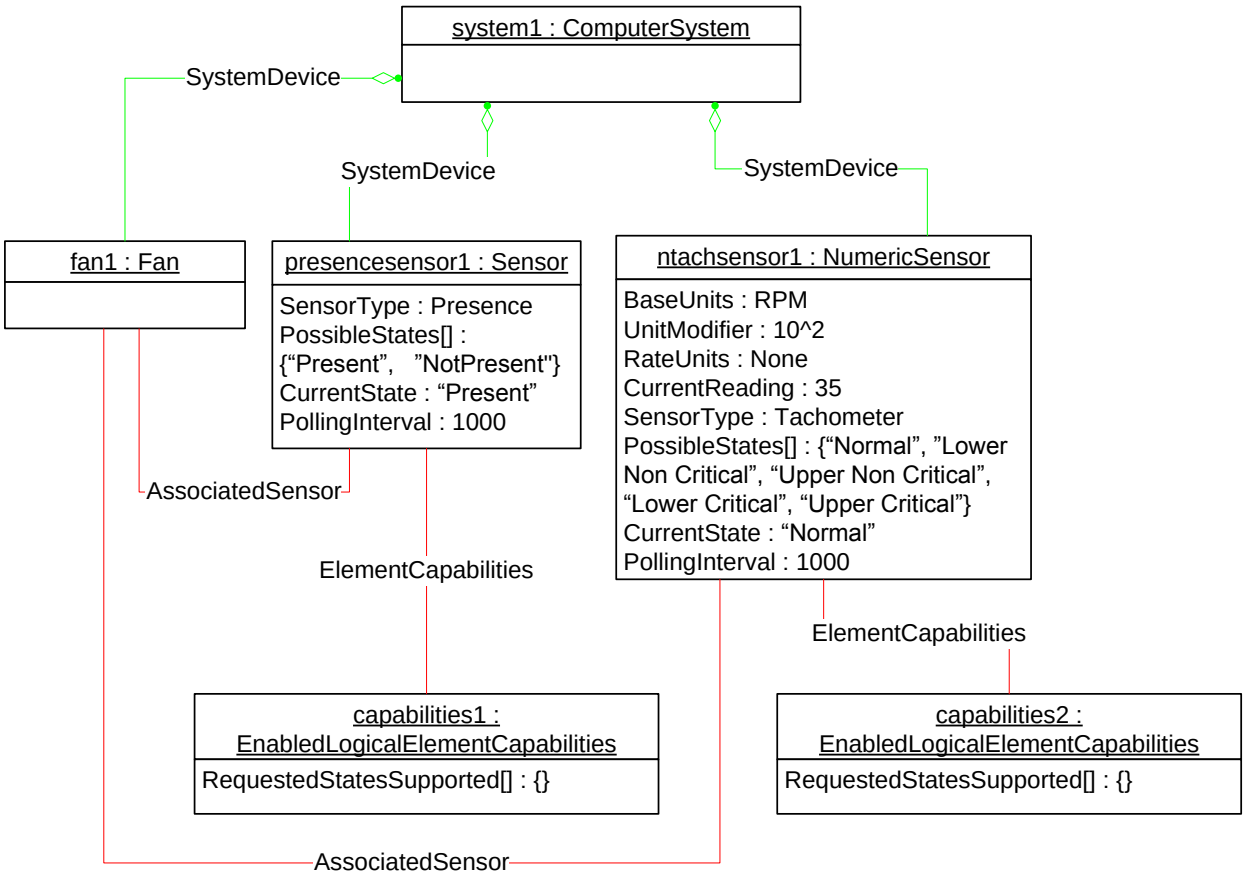

Figure 2 – Sensors Profile: Object diagram

9.2 Show all current states of the monitored devices in the computer system

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

9.3 Find sensors associated with a specific component

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

9.4 Change upper non-critical threshold of numeric sensor

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

9.5 Reset sensor to threshold defaults

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

9.6 Determining whether ElementName can be modified

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

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

9.7 Determining whether state management is supported

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

9.8 Find inlet temperature sensor instance

Enumerate the instances of CIM_NumericSensor. Determine whether the SensorContext property contains a value of “DMTF:Inlet” and the SensorType has value 2 (Temperature) for one of the instance of CIM_NumericSensor. If so, that instance represents the sensor for the inlet.

10 CIM Elements

Table 14 shows the list of CIM Elements for this profile and details their requirements. The implementation requirements for the classes and properties described in this section are defined in 7, “Implementation Requirements”.

Table 14 – CIM Elements: Sensors Profile 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		

10.1 CIM_Sensor

CIM_Sensor is used to represent a discrete sensor. The CIM_Sensor class is mandatory if the CIM_NumericSensor class is not implemented. Table 15 provides information about the properties of CIM_Sensor.

Table 15 – 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.
SensorContext	Optional	See 7.20.
RequestedState	Mandatory	See 7.14.
OperationalStatus	Mandatory	None
HealthState	Mandatory	None
RequestStateChange( )	Conditional	See 8.1.

10.2 CIM_NumericSensor

CIM_NumericSensor is used to represent an analog sensor. The CIM_NumericSensor class is mandatory when the CIM_Sensor class is not implemented. Table 16 provides information about the properties of CIM_NumericSensor.

Table 16 – 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.
SensorContext	Optional	See 7.20.
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.

10.3 CIM_EnabledLogicalElementCapabilities

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

Table 17 provides information about the properties of CIM_EnabledLogicalElementCapabilities.

Table 17 – 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.

10.4 CIM_ElementCapabilities

CIM_ElementCapabilities is used to associate CIM_Sensor or CIM_NumericSensor with an instance of CIM_EnabledLogicalElementCapabilities that describes the capabilities of CIM_Sensor or CIM_NumericSensor. Table 18 provides information about the properties of CIM_ElementCapabilities.

Table 18 – Class: CIM_ElementCapabilities

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

10.5 CIM_SystemDevice

CIM_SystemDevice is used to associate the instance of CIM_Sensor or CIM_NumericSensor with the instance of CIM_ComputerSystem of which the CIM_Sensor instance is a member. Table 19 provides information about the properties of CIM_SystemDevice.

Table 19 – 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.

10.6 CIM_AssociatedSensor

CIM_AssociatedSensor is used to associate the instance of CIM_Sensor or CIM_NumericSensor with the instance of a subclass of CIM_ManagedElement. Table 20 provides information about the properties of CIM_AssociatedSensor.

Table 20 – 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.

10.7 CIM_RegisteredProfile

CIM_RegisteredProfile is defined by the Profile Registration Profile. The requirements denoted in Table 21 are in addition to those mandated by the Profile Registration Profile.

Table 21 – 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 “1.1.0”.
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

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

ANNEX A
(informative)

Change log

Version	Date	Description
1.0.0	2007-11-06	
1.0.1	2008-09-25	Errata 1.0.1
1.0.2	2009-10-05	Errata 1.0.2
1.1.0	2012-08-07	DMTF Standard