Document Number: DSP1054

Date: 2010-05-20

Version: 1.1.0

Indications Profile

Document Type: Specification

Document Status: DMTF Standard

Document Language: en-US

CONTENTS

Figures

Tables

Foreword

The *Indications Profile* (DSP1054) was prepared by the DMTF WBEM Infrastructure Modeling Working Group.

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

Acknowledgments

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

Editor:

- Andreas Maier, IBM

Contributors:

- Jim Davis, WBEM Solutions (former editor)
- Steve Hand, Symantec (former editor)
- Jon Hass, Dell (former editor)
- Michael Johanssen, IBM
- David Judkovics, IBM (former editor)
- Aaron Merkin, IBM (former editor)
- Venkat Puvvada, IBM
- Karl Schopmeyer, DMTF Fellow
- Hemal Shah, Broadcom (former editor)

Introduction

The information in this specification should be sufficient for a provider or consumer of this data to unambiguously identify the classes, properties, methods, and values that shall be instantiated to subscribe, advertise, produce, or consume an indication using the DMTF Common Information Model (CIM) Schema.

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

Document Conventions

Typographical Conventions

Any text in this document is in normal text font, with the following exceptions:

- References to clause names use normal text font; if they consist of more than one word, the clause name is quoted using double quotes, such as in "CIM elements".
- Important terms that are used for the first time are marked in *italics*.
- The usage of terms link to the term definition defined in the "Terms and definitions" clause, enabling easy navigation to the term definition.
- ABNF rules are in `monospaced font`.

ABNF Usage Conventions

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

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

Deprecated Material

Deprecated material is not recommended for use in new development efforts. Existing and new implementations may use this material, but they shall move to the newer approach as soon as possible. An implementation of this profile in a CIM server shall use any deprecated material as if it was not deprecated, in order to achieve backwards compatibility for clients. Although implementations of clients may use deprecated material, it is recommended that they use the newer approach instead.

The following typographical convention indicates deprecated material:

DEPRECATED

Deprecated material appears here.

DEPRECATED

In places where this typographical convention cannot be used (for example, tables or figures), the "DEPRECATED" label is used alone.

Experimental Material

Experimental material has yet to receive sufficient review to satisfy the adoption requirements set forth by the DMTF. Experimental material is included in this document as an aid to implementers who are

interested in likely future developments. Experimental material may change as implementation experience is gained. It is likely that experimental material will be included in an upcoming revision of the specification. Until that time, experimental material is purely informational.

The following typographical convention indicates experimental material:

EXPERIMENTAL

Experimental material appears here.

EXPERIMENTAL

In places where this typographical convention cannot be used (for example, tables or figures), the "EXPERIMENTAL" label is used alone.

299 # Indications Profile

300 ## 1 Scope

301 The *Indications Profile* defines the CIM elements that are used to subscribe for indications of unsolicited
302 events, to advertise the possible indications, and to represent indications used to report events in a
303 managed system.

304 ## 2 Normative References

305 The following referenced documents are indispensable for the application of this document. For dated or
306 versioned references, only the edition cited (including any corrigenda or DMTF update versions) applies.
307 For undated and unversioned references, the latest published edition of the referenced document
308 (including any corrigenda or DMTF update versions) applies.

309 DMTF DSP0004, *CIM Infrastructure Specification 2.5*,
310 http://www.dmtf.org/standards/published_documents/DSP0004_2.5.pdf

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

313 DMTF DSP0228, *Message Registry XML Schema 1.1*,
314 http://schemas.dmtf.org/wbem/messageregistry/1/dsp0228_1.1.xsd

315 DMTF DSP0207, *WBEM URI Mapping Specification 1.0*,
316 http://www.dmtf.org/standards/published_documents/DSP0207_1.0.pdf

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

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

321 IETF RFC3986, *Uniform Resource Identifier (URI): Generic Syntax, January 2005*,
322 http://tools.ietf.org/html/rfc3986

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

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

327 ## 3 Terms and Definitions

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

3.1 General

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, Part2, 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, Part2, Annex H specifies additional alternatives. Occurrences of such additional alternatives shall be interpreted in their normal English meaning.

The terms "clause", "subclause", "paragraph", "annex" in this document are to be interpreted as described in ISO/IEC Directives, Part2, Clause 5.

The terms "normative" and "informative" in this document are to be interpreted as described in ISO/IEC Directives, Part2, Clause 3. In this document, clauses, subclauses or annexes indicated with "(informative)" as well as notes and examples do not contain normative content.

The terms defined in DSP0004, DSP0200 and DSP1001 apply to this document.

3.1
bulk subscription
an indication subscription to a filter collection that includes more than one indication filter

3.2
client
a WBEM client that exploits applicable portions of this profile

3.3
dynamic filter
an instance of CIM_IndicationFilter whose lifecycle is controlled by a client

3.4
event
the occurrence of a phenomenon of interest to a management application
Events are not published in CIM directly but may be represented by a model change or the instantiation of a CIM_Indication subclass.

3.5
implementation
a WBEM server that implements applicable portions of this profile

3.6
indication
a special kind of class that expresses the notification about an event that occurred
For a complete definition, see DSP0004. In addition, the indication may only represent an aspect of the event and not the entire event. Multiple indications may be communicated for a specific event.

3.7
indication filter
a logical construct that specifies a filter on indications, used to control whether indications are delivered to a subscriber.

3.8
indication service
a functionality of a WBEM server for indication related processing, including handling of subscriptions and delivery of indications to a WBEM listener.

3.9
listener
a WBEM listener that implements applicable portions of this profile.

3.10
query
a filter to constrain the events for which indications are generated.

3.11
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.

3.12
static filter
an instance of CIM_IndicationFilter whose lifecycle is controlled by an implementation.

3.13
subscribe
the mechanism whereby a client registers for delivery of indications.

3.14
WBEM client
a CIM client (see DSP0004) that supports a WBEM protocol. A WBEM client originates operations for processing by a WBEM server. This definition does not imply any particular implementation architecture or scope, such as a client library component or an entire management application.

3.15
WBEM listener
a CIM listener (see DSP0004) that supports a WBEM protocol. A WBEM listener processes indications originated by a WBEM server. This definition does not imply any particular implementation architecture or scope, such as a standalone demon or an entire management application.

3.16
WBEM server
a CIM server (see DSP0004) that supports a WBEM protocol. A WBEM server processes operations originated by a WBEM client, and originates indications for processing by a WBEM listener. This definition does not imply any particular implementation architecture, such as a separation into a CIMOM and provider components.

4 Symbols and Abbreviated Terms

4.1
CQL
CIM Query Language

4.2
QoS
Quality of service

4.3
URI
Uniform Resource Identifier

4.4
WBEM
Web Based Enterprise Management

5 Synopsis

Profile Name: Indications

Version: 1.1.0

Organization: DMTF

CIM Schema Version: 2.24

Central Class: CIM_IndicationService

Scoping Class: CIM_System

The *Indications Profile* extends the management capabilities defined in referencing profiles by adding the capability to subscribe for indications of unsolicited events, and to advertise the possible indications. The *Indications Profile* defines the content of indications from autonomous and component profiles implemented by CIM-based management instrumentation.

The central instance of this profile shall be an instance of CIM_IndicationService. The scoping instance shall be the instance of CIM_System with which the central instance is associated through CIM_HostedService.

Table 1 identifies profiles that are referenced by this profile.

Table 1 – Related Profiles

Profile name	Organization	Version	Relationship	Description
Profile Registration	DMTF	1.0	Mandatory	Registration of implementations of this profile.

6 Description

The *Indications Profile* describes the necessary properties and methods to describe the indications supported by managed elements and how a client subscribes a listener to those indications.

6.1 Overview of Profile Elements

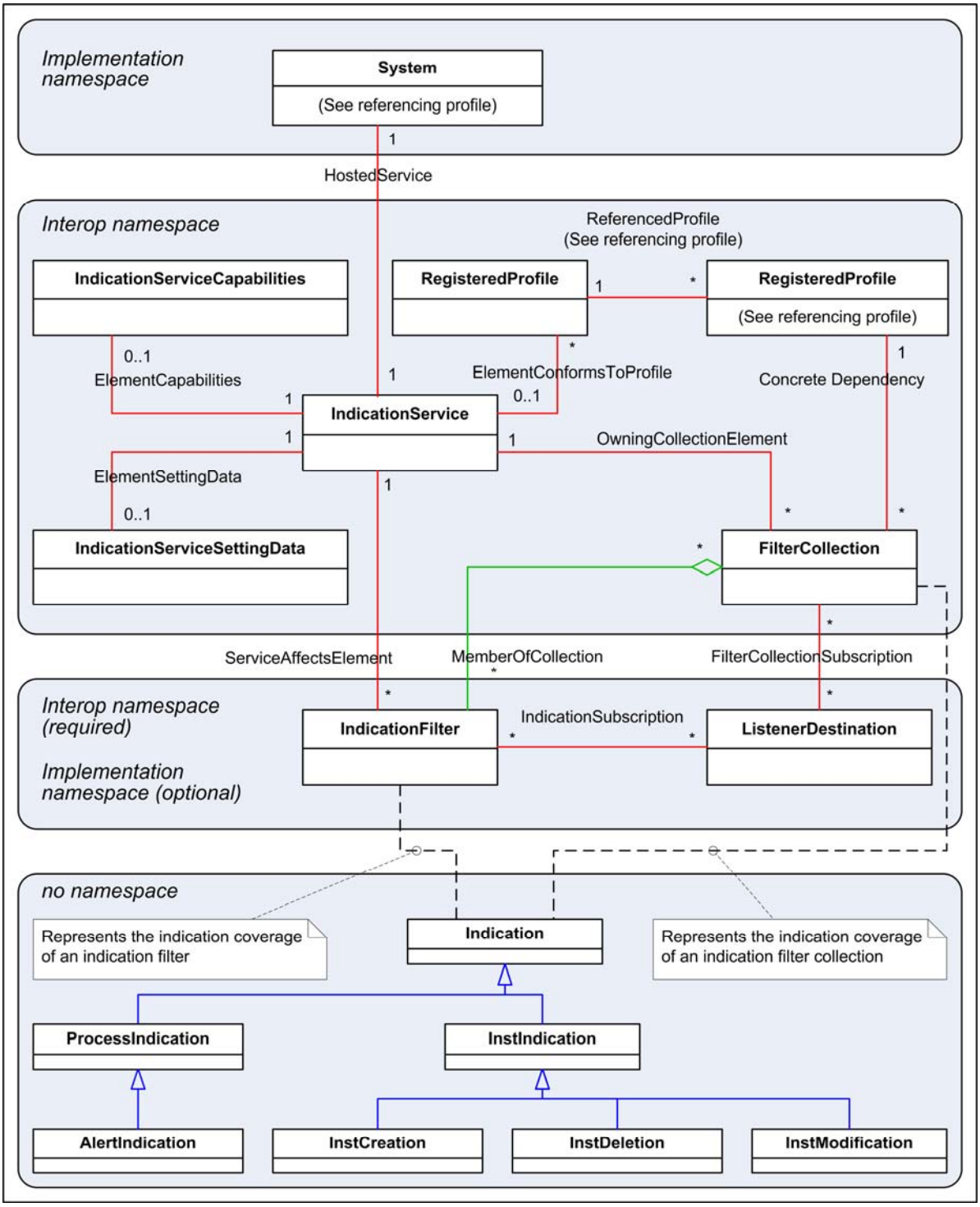
An event is some phenomenon of interest. An indication is an observation of characteristics of that event. For example, an event could be the fact that your house caught fire. An indication could report the fact that smoke or heat is observed; as the observer knows, smoke and heat are characteristics of fire. Alternatively, the indication may report that your house has caught fire.

Because CIM reports many characteristics of management elements in several classes and an event is likely to change several instances and properties, a change to any instance reports some of the characteristics of the event. As such, any given lifecycle indication reports observations.

CIM_AlertIndication instances are capable to reporting the event directly whether or not any characteristics of the event are modeled by an implementation. As such, a CIM_AlertIndication instance can report the event directly, but may not be able to convey any observations of the effect of the event.

448 Figure 1 represents the UML class diagram for the *Indications Profile*. For better clarity and
449 understanding, see DSP1033 for information about profile registration and namespaces.

450 For simplicity, the *CIM_* prefix has been removed from the names of the classes in Figure 1.



451

452 **Figure 1 – Indications Profile: Class Diagram**

CIM_IndicationFilter, CIM_FilterCollection, and CIM_ListenerDestination are instantiated in the Interop namespace. Creating the CIM_IndicationFilter, CIM_FilterCollection, and CIM_ListenerDestination instances in the Interop namespace (see DSP1033) makes it easier for clients to discover filters, collections of filters, and existing listener destinations that have been instantiated or are available.

CIM_IndicationService represents an indication service.

CIM_IndicationServiceCapabilities is an optional element that represents the capabilities of the CIM_IndicationService.

CIM_IndicationServiceSettingData is an optional element that is used to model the initial configuration of the CIM_IndicationService.

A CIM_IndicationFilter instance represents the potential of an implementation to produce an indication as described by the filter's query. The filter's query logically selects a particular modeled change, such as the creation of a CIM_AlertIndication or a change to the existing instance, from a population of all such changes. It appears to an observer that the implementation is monitoring all changes all the time. The lifecycle of CIM_IndicationFilter instances is controlled by either the implementation (static filters) or a WBEM client (dynamic filters) (see 6.3).

CIM_FilterCollection is used to describe a collection of filters supported in the context of a given profile (see 6.4).

CIM_ListenerDestination represents the location and method of delivering an indication to the client that may be subscribed to one or more indication filters. The Destination address in the CIM_ListenerDestination may be different than the network address of the WBEM client that created the subscription.

CIM_IndicationSubscription represents the request that indications described by an IndicationFilter or inferred by IndicationFilterCollection are delivered to a particular ListenerDestination.

CIM_FilterCollectionSubscription represents an active subscription of a destination (represented by CIM_ListenerDestination) to a collection of indication filters (represented by CIM_FilterCollection).

CIM_ConcreteDependency is used to scope instances of CIM_FilterCollection with instances of CIM_RegisteredProfile that identify the profile that provides context to the indication filters.

CIM_MemberOfCollection may be used to aggregate instances of CIM_IndicationFilter into one or more instances of CIM_FilterCollection.

CIM_OwningCollectionElement is used to scope instances of CIM_FilterCollection to the instance of CIM_IndicationService.

6.2 Client Indication Subscriptions

Using the behavior defined in the *Indications Profile*, WBEM clients are able to have indications from managed elements delivered to listeners by subscribing to one or more indication filters (which define query strings that select specific instances of subclasses of CIM_Indication).

6.2.1 Creating a Subscription

A WBEM client implements three steps to subscribe for indications:

1. Determine if there is an existing indication filter for the subscription. The indication filter may be explicitly modeled with an instance of CIM_IndicationFilter or implicitly represented by a CIM_FilterCollection that is defined to contain the indication filter. If an appropriate indication filter does not exist, and dynamic filters are supported, the WBEM client can create a dynamic filter.

2. Determine if the desired destination is already covered by looking for an instance of CIM_ListenerDestination that represents the destination. If one does not exist, the WBEM client may create one.

3. Create an instance of CIM_IndicationSubscription or CIM_FilterCollectionSubscription between the CIM_ListenerDestination and CIM_IndicationFilter or CIM_FilterCollection.

6.2.2 Bulk Subscriptions

A bulk subscription is a single subscription that encompasses one or more indication filters. Bulk subscriptions are implemented as an instance of CIM_FilterCollectionSubscription that associates an instance of CIM_ListenerDestination to an instance of CIM_FilterCollection. Subscribing to a filter collection is equivalent to individually subscribing to each indication filter in the collection and results in an indication being sent for every indication filter triggered by an event.

6.2.3 Recursive Subscriptions

An instance of CIM_FilterCollection implicitly contains indication filters that may be represented explicitly by instances of CIM_IndicationFilter. An instance of CIM_FilterCollection may contain additional CIM_FilterCollection instances. Subscription to a CIM_FilterCollection instance is interpreted as a single subscription to all contained indication filters and all contained instances of CIM_FilterCollection. Thus, if the same destination is explicitly subscribed to an instance of CIM_FilterCollection and is also explicitly subscribed to a contained instance of CIM_IndicationFilter or CIM_FilterCollection, the destination can receive duplicate notifications.

6.2.4 Subscriptions whose Filter Semantics Overlap

The same indication destination may be represented with more than one instance of CIM_ListenerDestination. The filter semantics between two subscriptions may overlap. The same indication filter may be represented multiple times. It may be represented explicitly by more than one instance of CIM_IndicationFilter or implicitly by one or more CIM_FilterCollection instances. This potential overlap makes it possible for more than one subscription to cause a particular indication to be delivered to a particular destination. The implementation does not perform any crosschecking to prevent the delivery of overlapping indications. Thus, the same indication can be produced from multiple indication filters. Therefore, it is the responsibility of a WBEM client to ensure that the subscriptions it creates do not result in overlapping filters for the same destination. .

6.2.5 Dynamic Contents of Filter Collections

A subscription to a CIM_FilterCollection instance is interpreted as a subscription to the filters contained within the collection. Although the indication filters implicitly contained in the collection do not change, it is possible that the indication filters explicitly contained (CIM_IndicationFilter or nested CIM_FilterCollection instances) may change. A snapshot of the contained filters at the time of the creation of the subscription is not maintained. Therefore, as the contents of the CIM_FilterCollection instance change, the set of filters to which the subscription actually applies may change.

6.3 Indication Filters

The CIM_IndicationFilter class represents a filter for selecting indications and contains a query string that defines selection criteria for events. Indication filters are used to identify the events created by managed elements and delivered by the implementation to the listener. The lifecycle of any filters can be controlled by either the implementation (static filters) or by a WBEM client (dynamic filters).

6.3.1 Filter Query

Filters identify the type of event to listen for and the CIM elements to be included in the indication delivered to any subscribed listeners. Filters are specified in the form of a query string that is contained in the Query property of a CIM_IndicationFilter instance.

The query defines the model changes or events that are being listened for. The query may define the model properties sent with the indication. A query also defines the source classes for the properties and what logic is used to combine the instances. A query is defined using the rules of a query language, like CIM Query Language (CQL). Profiles that define indications specify the exact string that represents the filter query.

Following are examples of a properly formatted CQL filter query:

EXAMPLE 1: “SELECT * FROM CIM_AlertIndication” — This query statement specifies that all supported properties of the CIM_AlertIndication instance can be delivered to listeners that are subscribed to this indication when such an event occurs.

EXAMPLE 2: “SELECT * FROM CIM_InstCreation WHERE SourceInstance ISA CIM_StorageVolume” — This query statement specifies that all supported properties of the CIM_InstCreation instance can be delivered to listeners and the CIM_InstCreation instance shall be delivered when the value of the SourceInstance property is an instance of CIM_StorageVolume.

6.3.2 Static Filters

Static filters are instances of CIM_IndicationFilter that are instantiated by an implementation. Static filters represent the events for which an implementation is capable of generating indications. These static filters enable a WBEM client to discover the supported indications of a given profile.

- **Mandatory Indication Filter**

  An indication filter defined in a profile as a mandatory indication filter is required to be supported if at least one indication filter defined in the profile is supported.

- **Optional Indication Filter**

  An indication filter defined in a profile as an optional indication filter may be supported.

- **Conditional Indication Filter**

  An indication filter defined in a profile as a conditional indication filter is supported if certain conditions are satisfied.

- **Vendor-Defined Indication Filter**

  An implementation may support instances of CIM_IndicationFilter that are not defined by a profile.

6.3.3 Dynamic Filters

Dynamic filters are instances of CIM_IndicationFilter that are created and deleted by a WBEM client and maintained by the implementation. Dynamic filters enable a listener to receive only the indications of interest. However, dynamic filters depend on the implementation being able to interpret the filter created by the WBEM client. Not all implementations, especially footprint-sensitive implementations, can act on the query defined in the filter.

While dynamic filters may be supported by an implementation, WBEM clients should first look for an existing instance of CIM_IndicationFilter that satisfies a need before attempting to create a dynamic filter. Adding unnecessary additional filters may adversely affect the performance of indication delivery by the implementation.

Finally, WBEM clients should check the CIM_IndicationService.FilterCreationEnabled property value to determine if the implementation supports dynamic filters before attempting the CreateInstance operation to create the filter (see 9.2 for this use case). If the property value is False, the implementation does not support dynamic filters and thus filter creation or deletion by a WBEM client.

6.4 Filter Collections

This clause describes filter collections in general and the three specific types of collections.

6.4.1 General

A filter collection comprises indication filters and other filter collections. Filter collections are represented by instances of CIM_FilterCollection, which is derived from CIM_Collection and inherits the CIM_Collection behavior.

A WBEM client may subscribe a listener to a filter collection directly. A subscription to a filter collection is recursively a subscription to all of the indication filters defined in the collection and any aggregated filter collections. An indication filter that is contained in a collection need not be explicitly modeled with an instance of CIM_IndicationFilter and associated through an instance of CIM_MemberOfCollection to the CIM_FilterCollection instance for the listener to receive indications matching the filter. If a listener is subscribed to a filter collection, for a given event the listener can receive a discrete indication for each indication filter in the collection the event matches.

Profiles may define multiple types of filter collections: mandatory, conditional, optional, and additional profile specific. Each filter collection can be defined to include one or more indication filters. If an implementation supports at least one indication that satisfies a filter contained in a collection, the collection can be instantiated.

Filter collections defined in a profile are associated with the instance of CIM_RegisteredProfile that represents the profile through an instance of CIM_ConcreteDependency. An instance of CIM_FilterCollection is associated with the instance of CIM_IndicationService through an instance of CIM_OwningCollectionElement.

The instances of CIM_FilterCollection are associated with zero or more instances of CIM_IndicationFilter by using the CIM_MemberOfCollection association to represent the collection of filters supported in the context of the associated CIM_RegisteredProfile.

6.5 When to Instantiate CIM_IndicationFilter

To accommodate implementation footprint concerns about the cost of instantiating all of the potential instances of CIM_IndicationFilter, the following approach is available to reduce the number of indication filters instantiated. This approach applies to mandatory, conditional and optional indication definitions in profiles.

Because a profile could define filter collections for the mandatory and conditional or optional indications defined in a profile, a WBEM client could subscribe a listener to a collection to receive all of the indications generated by the indication filters that are in that collection. In this case, it is not necessary to explicitly instantiate the instances of the CIM_IndicationFilter that represent each indication filter. This approach allows the actual instantiation of indication filter instances for mandatory and conditional or optional indications to be optional.

Following are two reasons to explicitly instantiate instances of CIM_IndicationFilter that represent static filters that are supported:

- To enable a WBEM client that does not have a priori knowledge of the indication filters specified by a profile to determine the indication filters supported by an implementation.

- To enable a WBEM client to subscribe a listener to individual filters instead of all filters in a collection.

An implementation may instantiate individual instances of CIM_IndicationFilter to satisfy the first goal without supporting individual subscription. The CIM_IndicationFilter.IndividualSubscriptionSupported property indicates whether subscription to the individual filter is supported.

Profiles may mandate specific instances of CIM_IndicationFilter and additionally mandate that individual subscription be supported. One reason for taking this approach is to enable WBEM clients to subscribe listeners to the most important events within the profile, which may be a subset of those supported. See ANNEX A for more information about specifying indication constraints in referencing profiles.

6.6 Listener Destinations

A few implementation paradigms may be supported by an implementation for management of listener destinations. An implementation may support listener destination management through creation and deletion of instances of CIM_ListenerDestination. Alternately, an implementation may statically create instances of CIM_ListenerDestination and support the specification of desired destinations through modification of the instance of CIM_ListenerDestination. An implementation may support a hybrid model, in which it allows creation, modification, and deletion of instances of CIM_ListenerDestination. If an implementation statically creates instances of CIM_ListenerDestination and supports WBEM client modification, the CIM_ListenerDestination.Destination property should be NULL until it is modified by a WBEM client. If a WBEM client wants to indicate that a CIM_ListenerDestination is no longer in use, and is available to be used to specify a new destination, the WBEM client should set the value of the CIM_ListenerDestination.Destination property to NULL.

6.7 Indication Service

CIM_IndicationService represents an indication service.

Various aspects of the indication service behavior are modeled, including the following:

- support for client-instantiated filters
- definition of indication delivery retry attempts
- definition of indication delivery retry intervals
- support for subscription removal action
- definition of the subscription removal time interval

6.7.1 CIM_IndicationService.FilterCreationEnabled

The FilterCreationEnabled property controls whether WBEM clients can create indication filters. If this value is set to False, WBEM clients cannot create indication filters and only the indication filters or filter collections provided by the implementation can be subscribed to. If this value is True, WBEM clients may attempt to create filters. The implementation shall reject the WBEM client filter creation attempt if the filter specified or filter creation in general cannot be supported. An implementation may preset this setting and not allow this value to be modified.

6.7.2 CIM_IndicationService.DeliveryRetryAttempts

The DeliveryRetryAttempts property defines the number of times that the indication service will try to deliver an indication to a particular listener destination. This value does not include the original delivery attempt; thus, if this value is set to 0, the indication service tries to deliver the indication only once. An implementation may preset this setting and not allow this value to be modified.

6.7.3 CIM_IndicationService.DeliveryRetryInterval

The DeliveryRetryInterval property defines the minimal time interval in seconds for the indication service to wait before delivering an indication to a particular listener destination that previously failed. The implementation may take longer due to QoS or other processing. An implementation may preset this setting and not allow this value to be modified.

6.7.4 CIM_IndicationService.SubscriptionRemovalAction

The SubscriptionRemovalAction property defines the removal action for subscriptions that have two failed indication deliveries without any successful indication deliveries in between if the time between the failed deliveries exceeded the timeout defined in the SubscriptionRemovalTimeInterval property. An implementation may preset this setting and not allow this value to be modified.

6.7.5 CIM_IndicationService.SubscriptionRemovalTimeInterval

The SubscriptionRemovalTimeInterval property defines the minimum time between two failed indication deliveries without any successful indication deliveries in between before the SubscriptionRemovalAction goes into effect.

6.7.6 CIM_IndicationServiceSettingData

The CIM_IndicationServiceSettingData class represents the configuration settings for the CIM_IndicationService class.

6.8 Indication Types and Processing

The two types of indications are

- lifecycle indications
- alert indications

Figure 2 depicts the indication class hierarchy. For simplicity, the *CIM_* prefix has been removed from the class names.

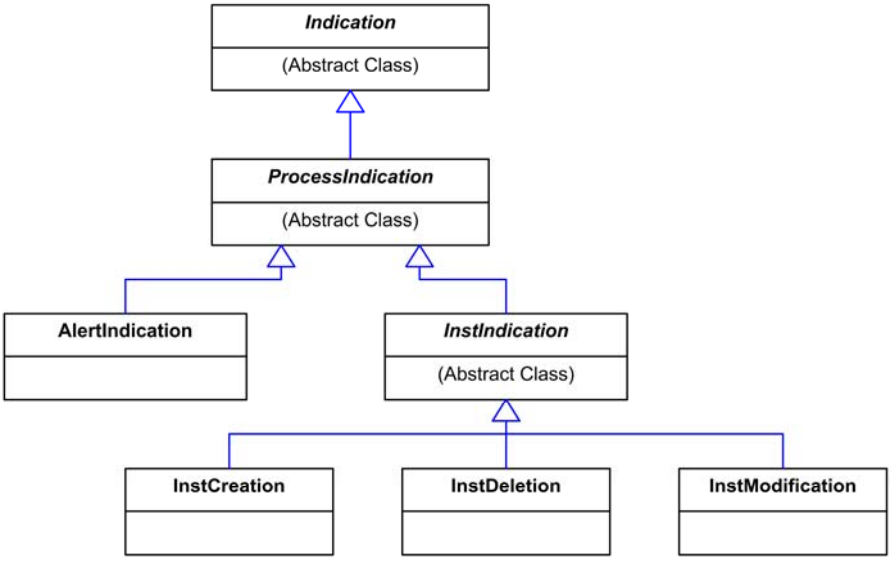


Figure 2 – Indication Class Diagram

686 ### 6.8.1 Lifecycle Indications

687 Lifecycle indications are indications that provide notification of changes in the lifecycle of CIM instances
688 and CIM class definitions. Only lifecycle indications related to changes in CIM instances are within the
689 scope of this profile. Lifecycle indications related to changes in CIM instances are reported using
690 instances of CIM_InstCreation, CIM_InstDeletion, or CIM_InstModification. They are used to convey
691 changes in the model that reflect observations of changes in the managed element.

692 ### 6.8.2 Alert Indications

693 Alert indications draw the attention of subscribing WBEM clients to the occurrence of an event. Alert
694 indications may describe aspects of an event that may or may not have other representation in CIM.

695 ## 6.9 Subscription Management Authorization

696 This profile makes no explicit provisions for managing the permissions of a WBEM client with respect to
697 its ability to create, modify, or delete indication subscriptions. Any coordination between WBEM clients or
698 access management to govern the ability of one WBEM client to make changes that affect the
699 subscriptions established by another WBEM client are outside the scope of this profile.

700 # 7 Implementation

701 This clause details the requirements related to the arrangement of instances and their properties for
702 implementations of this profile. Methods are listed in Clause 8 ("Methods") and properties are listed in
703 Clause 10 ("CIM Elements").

704 ## 7.1 CIM_IndicationService

705 CIM_IndicationService represents an indication service.

706 ### 7.1.1 General Requirements

707 One instance of CIM_IndicationService shall be instantiated in the Interop namespace.

708 Future versions of this profile may support more than one instance of CIM_IndicationService.

709 ### 7.1.2 Profile Default Configuration

710 To encourage consistent behavior across implementations of the indication service, a common default
711 configuration for each instance of CIM_IndicationService is defined. Unless the CIM_IndicationService
712 has been explicitly configured to behave differently, the following default values should be used for
713 selected properties of CIM_IndicationService:

714 - DeliveryRetryAttempts matches 3.
715 - DeliveryRetryInterval matches 20.
716 - SubscriptionRemovalAction matches 2 (Remove).
717 - SubscriptionRemovalTimeInterval matches 2,592,000.

718 NOTE: 2,592,000 seconds is equivalent to 30 days.

719 ## 7.2 CIM_IndicationServiceSettingData (Optional)

720 The CIM_IndicationServiceSettingData class is used for the initial configuration settings for the indication
721 service. An instance of CIM_IndicationServiceSettingData may be associated with the instance of
722 CIM_IndicationService through an instance of CIM_ElementSettingData.

7.3 Indication Filters

Support for an indication filter may be explicitly modeled with an instance of CIM_IndicationFilter. Support for an indication filter may be implicitly modeled by instantiating an instance of CIM_FilterCollection that is defined by a profile to contain the indication filter. Indication filters shall be defined as mandatory, optional, or conditional in a profile.

If an indication filter is defined as mandatory, the indication filter shall be supported if an implementation of a profile supports at least one indication filter defined in the profile.

If an indication filter is defined as optional or conditional, the indication filter may be supported.

7.4 CIM_IndicationFilter

CIM_IndicationFilter represents the potential of an implementation to produce a particular indication. The filter may also describe the model changes that can result in that indication. For lifecycle indications, the model change described in the query precedes the production of an indication communicating that change. For other types of indications, the model change may be the production of the indication instance itself.

7.4.1 General Requirements

On a CreateInstance operation request, if the specified CIM_IndicationFilter instance is supported by the implementation, it shall be created in the requested namespace. It shall also be created in the Interop namespace if the requested and Interop namespaces are different. All such instances shall have the same keys.

A creation of a CIM_IndicationFilter shall fail if its semantics are unable to be supported in the namespaces listed in SourceNamespaces property entries. If the operation fails, no instances shall be created.

Instantiation of a CIM_IndicationFilter may be initiated either by the implementation or by a WBEM client.

Each instance of CIM_IndicationFilter shall be associated with exactly one instance of CIM_IndicationService through an instance of CIM_ServiceAffectsElement.

One or more instances of CIM_IndicationFilter may be instantiated by either an implementation or by a WBEM client. Each instance of CIM_IndicationFilter shall be associated with exactly one instance of CIM_IndicationService through an instance of CIM_ServiceAffectsElement.

If the CIM_IndicationFilter.IndividualSubscriptionSupported property has the value True, the instance of CIM_IndicationFilter may be associated with one or more instances of CIM_ListenerDestination through an instance of CIM_IndicationSubscription. If the CIM_IndicationFilter.IndividualSubscriptionSupported property has the value False, the instance of CIM_IndicationFilter shall not be associated with any instances of CIM_ListenerDestination through an instance of CIM_IndicationSubscription.

Each instance of CIM_IndicationFilter may be associated with one or more instances of CIM_FilterCollection that represent vendor-supplied indications or other vendor-defined indication collections.

7.4.2 Indication Filter Validity

An instance of CIM_IndicationFilter shall be considered valid under the following conditions:

- The value of the QueryLanguage property identifies a query language supported by the indication service.
- The value of the Query property is well formed according to the supported query language. LifeCycle Indication Filters shall include a WHERE clause.

- The implementation is capable of producing indications that are selected by the filter.

7.4.3 Static Filter Creation

An implementation may instantiate instances of CIM_IndicationFilter for conditional, optional, or vendor-specific indications that are supported in the context of a profile implementation but that are beyond the scope of the indication requirements of that profile. If non-mandatory indications are supported, they shall be categorized into instances of CIM_FilterCollection that match the requirement from the profile (Mandatory, Conditional, Optional) or that are vendor-specific. CIM_FilterCollection instantiation requirements will be described in a future version of this document.

Autonomous profiles may define filters that include indications outside the immediate scope of the profile (for example, SELECT * FROM CIM_AlertIndication). An implementation may instantiate vendor-defined filters that are outside the scope of any particular profile.

If an instance of CIM_IndicationFilter represents a static filter that is mandatory in the defining profile, it shall be associated through an instance of CIM_MemberOfCollection with the instance of CIM_FilterCollection. If an instance of CIM_IndicationFilter represents a static filter that is optional or conditional in the defining profile, it shall be associated through an instance of CIM_MemberOfCollection with the instance of CIM_FilterCollection.

7.4.4 Dynamic Filter Creation

Constraints on the creation of dynamic filters are specified in 8.6.1.

Dynamic filters are instantiated by a WBEM client by using the intrinsic method CreateInstance. The management application populates the Query property with a properly formatted query per the requirements of the query language specified in the QueryLanguage property.

7.4.5 Subscribing to Dynamic Filters

WBEM clients subscribe listeners to dynamic filters by creating an instance of CIM_IndicationSubscription that references the CIM_IndicationFilter instance that represents the dynamic filter and an instance of CIM_ListenerDestination that represents the desired destination (see 8.9.1).

7.4.6 CIM_IndicationFilter.Query

When an instance of CIM_IndicationFilter is created, the Query property shall be populated with a properly formed query per the requirements of the query language identified in the QueryLanguage property.

EXPERIMENTAL

7.4.7 CIM_IndicationFilter.SourceNamespaces

For static filters, the SourceNamespaces property shall be formatted according to the format used by the implementation.

If an instance of CIM_IndicationFilter is implemented in the Interop namespace, the SourceNamespaces property shall contain the name of each namespace in which indications can be produced or that contains CIM_ManagedElement instances for which indications can be produced, where the indications match the filter specified by the CIM_IndicationFilter instance.

If an instance of CIM_IndicationFilter is implemented in an implementation namespace, the SourceNamespaces property does not need to be populated if the indication originates in the same namespace as the filter.

As part of defining dynamic filters, the SourceNamespaces array property is filled in by the WBEM client upon creation of the indication filter or upon subsequent modifications of the indication filter instance.

EXPERIMENTAL

7.4.8 CIM_IndicationFilter.Name

This subclause constrains the format of the value of the Name property, conformant to its definition in the CIM schema.

The value of the Name property in instances defined by referencing profiles shall be formatted as defined by the following ABNF rule:

```
OrgID ":" RegisteredName ":" UniqueID
```

Where:

`OrgID` shall identify the business entity owning the referencing profile. `OrgID` shall include a copyrighted, trademarked, or otherwise unique name that is owned by that business entity or that is a registered ID assigned to that business entity by a recognized global authority. In addition, to ensure uniqueness, `OrgID` shall not contain a colon (`:`).

For referencing profiles owned by DMTF, `OrgID` shall match `"DMTF"`.

`RegisteredName` shall be the registered name of the referencing profile, as defined by the value of its CIM_RegisteredProfile.RegisteredName property.

`UniqueID` shall uniquely identify the instance within the referencing profile.

DEPRECATED

For compatibility with version 1.0 of this profile, referencing profiles owned by business entities other than DMTF may in addition define values for the Name property that are formatted as defined by the following ABNF rule:

```
OrgID ":" UniqueID
```

Where:

`OrgID` is defined above in this subclause.

`UniqueID` shall uniquely identify the instance within the business entity owning the referencing profile.

Version 1.1 of this profile has deprecated this additional format.

DEPRECATED

7.5 CIM_ListenerDestination

CIM_ListenerDestination represents a destination for the delivery of indications.

7.5.1 General Requirements

On a create instance request, an instance of CIM_ListenerDestination shall be created in the namespace specified in the request. If the specified namespace is not the Interop namespace, an additional instance of CIM_ListenerDestination shall be created in the Interop namespace. Each such instance shall have the same keys.

Creation of a CIM_ListenerDestination shall fail if its semantics are unable to be supported in the Interop namespace or its creation namespace.

Instantiation of a CIM_ListenerDestination may be initiated either by the implementation or by a WBEM client.

Each instance of CIM_ListenerDestination shall be associated with exactly one instance of CIM_IndicationService through an instance of CIM_ServiceAffectsElement.

Any instance of CIM_ListenerDestination may be associated with one or more instances of CIM_IndicationFilter through an instance of CIM_IndicationSubscription, with one or more instances of CIM_FilterCollection through an instance of CIM_FilterCollectionSubscription, or both.

If an instance of CIM_ListenerDestination is not associated with any instance of CIM_IndicationFilter or CIM_FilterCollection, the WBEM client should reuse the instance of CIM_ListenerDestination and not create a new one.

7.5.2 CIM_ListenerDestination.Destination

If the value of the CIM_ListenerDestination.Destination property is not NULL, the property value shall be a valid IETF Uniform Resource Identifier value (as defined in RFC3986). The implementation shall reject a value that does not include the scheme, host and port as part of the URI Location.

7.5.3 CIM_ListenerDestination.PersistenceType

The PersistenceType property of a CIM_ListenerDestination instance describes the durability of the delivery destination description represented by that instance.

The property values shall be constrained to 3 (Transient), 2 (Permanent), and NULL.

A property value of NULL or 2 (Permanent) indicates that the delivery destination is permanent. Permanent delivery destinations are long-lived and are expected to be available for indication delivery. For example, a typical permanent delivery destination would be a system log file. An inability of an implementation to deliver indications to a listener described by a permanent delivery destination will be treated as an error condition by the implementation, as defined in 7.10.

A property value of 3 (Transient) indicates that the delivery destination is transient. Transient delivery destinations are short-lived and have less strong requirements (than permanent destinations) regarding their availability for indication delivery. For example, a typical transient delivery destination would be a task progress meter in a graphical management application. An inability of an implementation to deliver an indication to a WBEM listener described by a transient delivery destination will be handled by removing the delivery destination and its subscriptions from the implementation, as defined in 7.12.

7.6 CIM_FilterCollection

CIM_FilterCollection is used to define a collection of indication filters supported in the context of a particular profile or implementation.

Each instance of CIM_FilterCollection shall be instantiated in the Interop Namespace.

Creation of a CIM_FilterCollection shall fail if its semantics are unable to be supported in the Interop namespace.

Either a WBEM client or the implementation may create instances of CIM_FilterCollection.

7.6.1 Relationship with Indication Service

Every instance of CIM_FilterCollection shall be associated with exactly one instance of CIM_IndicationService through an instance of CIM_OwningCollectionElement.

7.6.2 Nested Filter Collections

An instance of CIM_FilterCollection may be associated with one or more instances of CIM_FilterCollection through an instance of CIM_MemberOfCollection.

7.6.3 Relationship with Registered Profile

Each instance of CIM_FilterCollection shall be associated with exactly one instance of CIM_RegisteredProfile through an instance of CIM_ConcreteDependency where the instance of CIM_RegisteredProfile represents the registration of the profile to which the indications pertain. This allows a WBEM client to discover all of the mandatory, optional, conditional, and vendor-specific indication filters supported by the implementation of a particular profile.

7.6.4 CIM_FilterCollection.CollectionName

This subclause constrains the format of the value of the CollectionName property, conformant to its definition in the CIM schema.

The value of the CollectionName property in instances defined by referencing profiles shall be formatted as defined by the following ABNF rule:

```
OrgID ":" RegisteredName ":" UniqueID
```

Where:

`OrgID` shall identify the business entity owning the referencing profile. `OrgID` shall include a copyrighted, trademarked, or otherwise unique name that is owned by that business entity or that is a registered ID assigned to that business entity by a recognized global authority. In addition, to ensure uniqueness, `OrgID` shall not contain a colon (`:`).

For referencing profiles owned by DMTF, `OrgID` shall match `"DMTF"`.

`RegisteredName` shall be the registered name of the referencing profile, as defined by the value of its CIM_RegisteredProfile.RegisteredName property.

`UniqueID` shall uniquely identify the instance within the referencing profile.

DEPRECATED

For compatibility with version 1.0 of this profile, referencing profiles owned by business entities other than DMTF may in addition define values for the Name property that are formatted as defined by the following ABNF rule:

```
OrgID ":" UniqueID
```

Where:

913 `OrgID` is defined above in this subclause.

914 `UniqueID` shall uniquely identify the instance within the business entity owning the referencing
915 profile.

916 Version 1.1 of this profile has deprecated this additional format.

917 **DEPRECATED**

---

918 7.7 WBEM Server Requirements

919 An implementation may support indications. If so, it shall meet the following requirements:

920 • instantiate a single instance of CIM_IndicationService

921 • support the indications of the *Indications Profile* as specified in the CIM Elements table in
922 Clause 10

923 • support the ability to subscribe for indications using the classes defined in the *Indications Profile*

924 • support indication filters in the Interop namespace

925 • support indications as defined in profiles that are advertised as implemented in the Interop
926 namespace

927 An implementation may support dynamic filters (instances of CIM_Indication filter or
928 CIM_FilterCollection).

929 7.8 CIM_IndicationSubscription

930 On a create instance request, if the corresponding CIM_IndicationSubscription instance is supported, it
931 shall be created in the requested namespace. It shall also be created in the Interop namespace if the
932 requested namespace and the Interop namespace are different. Additionally, for each source namespace
933 listed in the corresponding CIM_IndicationFilter instance found in the Interop namespace a corresponding
934 instance of CIM_IndicationSubscription should be instantiated between the corresponding
935 CIM_IndicationFilter instance in the source namespace and the associated CIM_ListenerDestination
936 instance in that same namespace if it exists.

937 A creation of a CIM_IndicationSubscription shall fail if its semantics are unable to be supported in the
938 Interop namespace or its creation namespace.

939 Instantiation of a CIM_IndicationSubscription may be initiated either by the implementation or by a WBEM
940 client.

941 7.8.1 CIM_IndicationSubscription.OnFatalErrorPolicy

942 A WBEM client uses the CIM_IndicationSubscription.OnFatalErrorPolicy property to define the desired
943 behavior for a subscription when a failure occurs that implies that some aspect of indication generation
944 processing or dispatch is no longer functioning and indications may be lost. A value of 4 (Remove)
945 requires that an implementation abide by the CIM_IndicationService.SubscriptionRemovalAction setting
946 (see 7.1) and behavior. The default value for this property should be 4 (Remove) if the WBEM client does
947 not specify a value.

948 7.8.2 CIM_IndicationSubscription.RepeatNotificationPolicy

949 The RepeatNotificationPolicy property of the CIM_IndicationSubscription class defines the desired
950 behavior for handling indications that report the occurrence of the same underlying event (for example,
951 the disk is still generating I/O errors and has not yet been repaired). This also includes multiple
952 indications that are generated from a single indication filter. Repeated indications are indications in which

all the indication instance property values are the same except for the IndicationIdentifier, IndicationTime, SequenceContext, and SequenceNumber properties.

The use of the RepeatNotificationCount, RepeatNotificationInterval, and RepeatNotificationGap properties defined in the CIM_IndicationSubscription class depends on the value of the RepeatNotificationPolicy property.

The RepeatNotificationPolicy may vary by implementation (or even IndicationFilter). However, it shall be specified on all subscriptions. The valid values are as follows:

- 2 (None)
- 3 (Suppress)
- 4 (Delay)

A profile may restrict these values further for any given indication filter, but it shall not expand the values to other policies due to interoperability constraints. For example, a profile may restrict InstCreation filters for CIM_ComputerSystem to 2 (None) and restrict InstModification filters on CIM_StorageVolume to Suppress or Delay. However, profiles shall not define Unknown as a valid setting for the RepeatNotificationPolicy property.

7.8.2.1 RepeatNotification Policy Property Value of 2 (None)

If the value of the RepeatNotificationPolicy property is 2 (None), special processing of repeat indications shall not be performed.

7.8.2.2 RepeatNotification Policy Property Value of 3 (Suppress)

If the value of the RepeatNotificationPolicy property is 3 (Suppress), indications are delivered up to the value of the RepeatNotificationCount property; after that, all subsequent indications are suppressed for the time interval defined in the RepeatNotificationInterval property. When the time interval expires, suppression expires. Any indication that matches the filter is included in the calculation of the indication count that is compared with the RepeatNotificationCount value. A new interval starts when the next indication for this event is received after the previous interval has expired.

7.8.2.3 RepeatNotification Policy Property Value of 4 (Delay)

If the value of the RepeatNotificationPolicy property is 4 (Delay) and an indication is generated, this indication shall be suppressed if, including this indication, RepeatNotificationCount or fewer indications for the same event have been generated during the time interval defined by RepeatNotificationInterval. If this indication is the RepeatNotificationCount + 1 indication instance generated, this indication shall be delivered and all subsequent indications for this event shall be ignored until the RepeatNotificationGap has elapsed. A RepeatNotificationInterval may not overlap a RepeatNotificationGap time interval.

7.9 CIM_FilterCollectionSubscription

On a create instance request, if the corresponding CIM_FilterCollectionSubscription instance is supported, it shall be created in the creation namespace, and if different, the Interop namespace.

A creation of a CIM_FilterCollectionSubscription shall fail if its semantics are unable to be supported in the Interop namespace or its creation namespace.

Instantiation of a CIM_FilterCollectionSubscription may be initiated either by the implementation or by a WBEM client.

7.10 Indication Delivery

This subclause defines the delivery of indications from an implementation to a WBEM listener.

If sequence identifiers, as described in subclause 7.10.1, are implemented by the implementation and by the listener, indication delivery becomes more reliable in that unsuccessful deliveries can be retried by the implementation, lost and duplicate deliveries can be detected by the listener, and indications arriving out of order can be reordered by the listener to the original order.

Implementing sequence identifiers is optional for an implementation and for a listener.

An implementation of sequence identifiers in an implementation can be discovered by the sequence identifier property of the capabilities class.

7.10.1 Sequence Identifier

This subclause defines the concepts of *sequence identifier value* and *sequence identifier lifetime*. It applies only if sequence identifiers are implemented.

The *sequence identifier value* of an indication is the combination of the SequenceContext and SequenceNumber property values of the CIM_Indication instance representing the indication, as defined in the CIM schema.

The *sequence identifier lifetime* of an indication service is a duration defined as follows:

sequence-identifier-lifetime = DeliveryRetryAttempts * DeliveryRetryInterval * 10

where DeliveryRetryAttempts and DeliveryRetryInterval are the values of the like-named properties of the CIM_IndicationService instance representing the indication service.

The sequence identifier value and sequence identifier lifetime enable a listener that implements sequence identifiers to efficiently detect lost, duplicate, and out-of-order deliveries sent from an implementation that also implements sequence identifiers, as described in the following subclauses.

7.10.2 WBEM Server Requirements

Indication delivery is based on a publish/subscribe event paradigm, where an implementation delivers indications to subscribed WBEM listeners. The indication delivery may fail for multiple reasons, including unavailability of the listener or network issues. This subclause describes requirements for the implementation that are related to the delivery of indications. The mechanism to deliver an indication and to determine success or failure of the delivery is protocol dependent. See the specifications of the protocols implemented for indication delivery.

When an indication subscription is disabled or has been removed, the implementation should discard any undelivered indications for that subscription. For example, this may happen when the implementation has queued indications for delivery retry and the subscription was removed by a WBEM client after a prior delivery attempt.

After an implementation has successfully delivered an indication to a listener, it shall not attempt to deliver that indication again to that listener.

If the attempt of an implementation to deliver an indication to a listener fails, the implementation shall retry the delivery as defined by the values of the DeliveryRetryAttempts and DeliveryRetryInterval properties of the CIM_IndicationService instance associated with the CIM_IndicationFilter or CIM_FilterCollection instance that caused the indication to be raised. If these retry attempts are exhausted for an indication delivery to a listener, that indication shall be considered unable to be delivered to that listener. See 7.10.2.1 for details on inability to deliver indications.

If an implementation implements sequence identifiers for its indication service, it shall limit the duration for which retries of failed indication deliveries are attempted after the initial delivery attempt to that listener, to the sequence identifier lifetime of that indication service. If this duration is exceeded for an indication whose delivery is being retried, that indication shall be considered unable to be delivered to that listener, even if the DeliveryRetryAttempts and DeliveryRetryInterval properties would otherwise indicate that further retries should be attempted. See 7.10.2.1 for details on inability to deliver indications. This additional time limitation for delivery retries ensures that any sequence identifier value from a particular indication service is guaranteed to be unique within the sequence identifier lifetime of that indication service. DeliveryRetryInterval defines a minimum time for the retry interval; so using these two properties alone does not establish an upper limit for the retry duration. The same indication may need to be delivered to multiple listeners; so each such delivery is handled separately.

7.10.2.1 Inability to Deliver Indications

This subclause defines the handling within an implementation if an indication has been considered unable to be delivered to a WBEM listener.

If the delivery destination describing that listener is permanent (see 7.5.3 for details), the implementation shall record an error and shall no longer attempt to deliver that indication to that listener (that is, discard it). This action does not modify the delivery destination and any of its subscriptions.

If the delivery destination describing that listener is transient (see 7.5.3 for details), the implementation shall record an error and shall no longer attempt to deliver that indication to that listener (that is, discard it). The delivery destination and its subscriptions are removed from the implementation as described in 7.12.

7.10.3 WBEM Listener Requirements

7.10.3.1 General

A WBEM listener that implements sequence identifiers shall keep track of each distinct sequence identifier value of any indications received from a particular indication service for the duration of the sequence identifier lifetime of that indication service, counting from the last time that sequence identifier value was seen in a received indication from that indication service. In other words, if the same sequence identifier value is used by two different indication services (for example, in two different implementations), the listener will keep track of them independently, and if the same sequence identifier value is used in different indications from the same indication service, receiving the second one within the lifetime of the first one will restart the lifetime for that sequence identifier value.

After the lifetime of a sequence identifier value expires, the listener should discard the knowledge about that sequence identifier value from that indication service. After the knowledge about a sequence identifier value for an indication service has been discarded by the listener, a new usage of that sequence identifier value in an indication from that indication service shall be treated by the listener like a new, unknown sequence identifier value from that indication service.

Keeping track of sequence identifier values in listeners enables the detection of lost and duplicate deliveries, and the detection and re-ordering of indications arriving out of order, as described in 7.10.3.4. Discarding the knowledge about sequence identifier values minimizes the resource requirements of the listener.

A listener that does not implement sequence identifiers may ignore the values of the SequenceContext and SequenceNumber properties in any received CIM_Indication instances.

7.10.3.2 Lost Indications

If a WBEM listener receives the indication, it caches the indication for a period defined by sequence identifier lifetime as defined in 7.10.3.1. From the value of the SequenceNumber property of the last indication received, a listener can infer the SequenceNumber of the next indication to be received (by incrementing the SequenceNumber by 1, wrapping to an initial value of 0 when the maximum limit has been reached and the SequenceContext has not changed). If during processing of the next received indication its CIM_Indication.SequenceNumber does not match the predicted value the listener shall consider the indications as a candidate for a lost indication. After waiting for the sequence identifier lifetime period of the last received indication, the listener shall conclude that the missing indication is lost.

7.10.3.3 Duplicate Indications

When an indication is successfully received by a WBEM listener its SequenceIdentifier is cached for a period defined by sequence identifier lifetime as defined in 7.10.3.1. Any additional indications received with the same SequenceIdentifier shall be considered duplicates.

7.10.3.4 Out-of-Order Indications

When a WBEM listener receives the indication, it caches the indication for a period defined by sequence identifier lifetime as defined in 7.10.3.1. From the SequenceNumber of the last indication received, a listener can infer the SequenceNumber of the next indication to be received (by incrementing the SequenceNumber by 1, wrapping to an initial value of 0 when the maximum limit has been reached and the SequenceContext has not changed).

If the SequenceIdentifier of the next received indication does not match the anticipated SequenceIdentifier, the listener can cache the missed anticipated SequenceIdentifier for a period defined by sequence identifier lifetime (as defined in 7.10.3.1) of the last received indication. If the anticipated SequenceIdentifier is not received during that period, the indications should be considered lost (see 7.10.3.2).

If the anticipated SequenceIdentifier is received during that period, the indication order shall be re-ordered using the SequenceNumber, such that the indications are processed in the order they were sent by the implementation. A listener that intends to re-establish the original order of indications before processing them needs to defer the processing of any indication that does not have the predicted sequence number, until the decision can be made as to whether the missing indications are lost, as described in this subclause.

7.11 Using Message Registries

A message registry is an XML document that contains entries that consist of standard message identifiers and static and dynamic message elements. An instance of CIM_AlertIndication may contain a standard message. The OwningEntity, MessageID, Message, and MessageArguments properties of the CIM_AlertIndication class are used to describe the content of an alert indication that is produced by instrumentation for a managed element. See DSP0228 for further provisions.

If an instance of CIM_AlertIndication contains a standard message, the following constraints shall be met:

- The MessageID property shall contain the message identifier from the registry.
- The OwningEntity property shall contain the identifier of the organization that defined the registry.
- The MessageArguments property shall contain the dynamic content of the message as defined by the message registry. The absolute ordering of the dynamic content shall be maintained.
- The Message property may contain the formatted message from the registry.

7.12 Indication Subscription Removal

The implementation may remove an indication subscription if the delivery destination (that is, CIM_ListenerDestination.Destination) cannot be reached within the number of delivery retry attempts and the retry interval specified in the CIM_IndicationServiceSettingData instance's DeliveryRetryAttempts and DeliveryRetryInterval properties. The removal of an indication subscription is governed by the CIM_IndicationService.SubscriptionRemovalAction property value. If the SubscriptionRemovalAction property has a value of 2 (Remove), the subscription shall be removed after two failed indication deliveries occur without any successful indication deliveries in between if the time between the deliveries exceeds the timeout specified in the CIM_IndicationService.SubscriptionRemovalTimeInterval property.

A WBEM client may remove an indication subscription by performing a DeleteInstance operation on the association instance created to activate the indication subscription (that is, the instance of CIM_IndicationSubscription or CIM_FilterCollectionSubscription). If there are no other subscriptions to this destination, the WBEM client may additionally remove the CIM_ListenerDestination that identified the indication delivery destination or leave that instance for future indication subscription.

7.13 Implementation of Profile Specifications

An implementation shall deliver all supported lifecycle indications to all listeners that are subscribed to filters that select the supported lifecycle indications.

An implementation shall deliver all supported alert indications to all listeners that are subscribed to filters that select the supported alert indications.

7.14 CIM_IndicationServiceCapabilities

An instance of CIM_IndicationServiceCapabilities shall be instantiated when the implementation supports the direct modification of any properties of the indication service. The CIM_IndicationServiceCapabilities instance shall be associated with the affected instance of CIM_IndicationService through an instance of CIM_ElementCapabilities. If the implementation does not support the direct modification of any properties on the indication service, the implementation may not instantiate an instance of CIM_IndicationServiceCapabilities. The absence of an instance of CIM_IndicationServiceCapabilities associated with the CIM_IndicationService indicates that modification of properties of the CIM_IndicationService by a WBEM client is not supported.

7.15 Indication.IndicationFilterName Property

At the time of the creation of an indication, an implementation may not have the information about the indication filters and/or filer collections that match the created indication. After the creation of the indication, the information about the indication filters and/or filter collections that matched the indication becomes known. Before the delivery of the indication, the information about all the matched indication filters shall be included in the IndicationFilterName property. The IndicationFilterName property contains the indication filter names (values of property CIM_IndicationFilter.Name) for the indication that matched the indication filters listed in this array. For each active subscription to each of the matched indication filters and/or filter collections, the indication shall be delivered. A WBEM client may use this property to match the indication received with semantics known a priori by the WBEM client. A profile ought to list the indications that an implementation can produce and why. A WBEM client uses this property to determine what indication was produced, as documented in the profile, and why.

If the IndicationFilter class is implemented, then the IndicationFilterName property of each instance of CIM_Indication shall contain the names of the indication filters that matched the indication. Otherwise, this property shall contain implementation specific name(s) that allow the listener to match the indication with the implementation-specific semantics.

7.16 Indications for the *Indications Profile*

This clause details the constraints for supporting indications specific to the *Indications Profile*.

7.16.1 Mandatory Indications

No mandatory indications are specified in this profile; therefore, there is no definition of a mandatory filter collection.

7.16.2 Conditional and Optional Indications

This clause describes the requirements for conditional and optional indications for implementations of the *Indications Profile*.

7.16.2.1 Conditional/Optional Filter Collection

There may be an instance of CIM_FilterCollection in which the CIM_FilterCollection.CollectionName property has the value "DMTF:Indications:Conditional/Optional".

7.16.2.2 Listener Destination Removal

There may be an indication filter as defined in this clause. Subscribers to this indication filter can be informed when a listener destination is deleted.

7.16.2.2.1 Indication Filter Name

The indication filter name shall be "DMTF:Indications:ListenerDestinationRemoval".

7.16.2.2.2 Filtered Events

The indication filter shall filter for notification of the deletion of instances of CIM_ListenerDestination.

7.16.2.2.3 Query

The CIM_IndicationFilter.Query property may have the value "SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_ListenerDestination".

7.16.2.3 Indication Subscription Removal

There may be an indication filter as defined in this clause.

Subscribers to this indication will be informed when a subscription is deleted. An indication will not be sent to the listeners who have been unsubscribed because the subscription is absent.

7.16.2.3.1 Indication Filter Name

The indication filter name shall be "DMTF:Indications:IndicationSubscriptionRemoval".

7.16.2.3.2 Filtered Events

The indication filter shall filter for notification of the deletion of instances of CIM_IndicationSubscription.

7.16.2.3.3 Query

The CIM_IndicationFilter.Query property may have the value "SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_IndicationSubscription".

7.16.2.4 Filter Collection Subscription Removal

There may be an indication filter as defined in this clause.

Subscribers to this indication will be informed when a subscription to a filter collection is deleted. An indication will not be sent to the listeners who have been unsubscribed because the subscription is absent.

7.16.2.4.1 Indication Filter Name

The indication filter name shall be "DMTF:Indications:FilterCollectionSubscriptionRemoval".

7.16.2.4.2 Filtered Events

The indication filter shall filter for notification of the deletion of instances of CIM_FilterCollectionSubscription.

7.16.2.4.3 Query

The CIM_IndicationFilter.Query property may have the value "SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FilterCollectionSubscription".

8 Methods

This section details the requirements for supporting intrinsic operations for the CIM elements defined by this profile. No extrinsic methods are defined by this profile.

8.1 Profile Conventions for Operations

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

The default list of operations is as follows:

- GetInstance( )
- Associators( )
- AssociatorNames( )
- References( )
- ReferenceNames( )
- EnumerateInstances( )
- EnumerateInstanceNames( )

8.2 CIM_HostedService

Table 2 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 2, all operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

1226 **Table 2 – Operations: CIM_HostedService**

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

1227 ## 8.3 CIM_IndicationService

1228 Table 3 lists implementation requirements for operations. If implemented, these operations shall be
1229 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 3, all operations in
1230 the default list in 8.1 shall be implemented as defined in DSP0200.

1231 NOTE: Related profiles may define additional requirements on operations for the profile class.

1232 **Table 3 – Operations: CIM_IndicationService**

Operation	Requirement	Messages
ModifyInstance	Conditional	See 8.3.1.

1233 ### 8.3.1 CIM_IndicationService — ModifyInstance

1234 This section details the requirements for the ModifyInstance operation applied to an instance of
1235 CIM_IndicationService.

1236 #### 8.3.1.1 General

1237 Support for the ModifyInstance operation is conditional. The ModifyInstance operation shall be supported
1238 for an instance of CIM_IndicationService if an instance of CIM_IndicationServiceCapabilities is associated
1239 with the CIM_IndicationService instance and at least one of the following properties of the
1240 CIM_IndicationServiceCapabilities instance has a value of True:

1241 - FilterCreationEnabledIsSettable
1242 - DeliveryRetryAttemptsIsSettable
1243 - DeliveryRetryIntervalIsSettable
1244 - SubscriptionRemovalActionIsSettable
1245 - SubscriptionRemovalTimeIntervalIsSettable

1246 #### 8.3.1.2 CIM_IndicationService.FilterCreationEnabled

1247 If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance
1248 and the FilterCreationEnabledIsSettable property of the CIM_IndicationServiceCapabilities instance has a
1249 value of True, the implementation shall allow the ModifyInstance operation to change the value of the
1250 FilterCreationEnabled property of the CIM_IndicationService instance.

1251 If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance
1252 and the FilterCreationEnabledIsSettable property of the CIM_IndicationServiceCapabilities instance has a
1253 value of False, the implementation shall not allow the ModifyInstance operation to change the value of the
1254 FilterCreationEnabled property of the CIM_IndicationService instance.

8.3.1.3 CIM_IndicationService.DeliveryRetryAttempts

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the DeliveryRetryAttemptsIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of True, the implementation shall allow the ModifyInstance operation to change the value of the DeliveryRetryAttempts property of the CIM_IndicationService instance.

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the DeliveryRetryAttemptsIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of False, the implementation shall not allow the ModifyInstance operation to change the value of the DeliveryRetryAttempts property of the CIM_IndicationService instance.

8.3.1.4 CIM_IndicationService.DeliveryRetryInterval

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the DeliveryRetryIntervalIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of True, the implementation shall allow the ModifyInstance operation to change the value of the DeliveryRetryInterval property of the CIM_IndicationService instance.

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the DeliveryRetryIntervalIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of False, the implementation shall not allow the ModifyInstance operation to change the value of the DeliveryRetryInterval property of the CIM_IndicationService instance.

8.3.1.5 CIM_IndicationService.SubscriptionRemovalAction

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the SubscriptionRemovalActionIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of True, the implementation shall allow the ModifyInstance operation to change the value of the SubscriptionRemovalAction property of the CIM_IndicationService instance.

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the SubscriptionRemovalActionIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of False, the implementation shall not allow the ModifyInstance operation to change the value of the SubscriptionRemovalAction property of the CIM_IndicationService instance.

8.3.1.6 CIM_IndicationService.SubscriptionRemovalTimeInterval

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the SubscriptionRemovalTimeIntervalIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of True, the implementation shall allow the ModifyInstance operation to change the value of the SubscriptionRemovalTimeInterval property of the CIM_IndicationService instance.

If an instance of CIM_IndicationServiceCapabilities is associated with the CIM_IndicationService instance and the SubscriptionRemovalTimeIntervalIsSettable property of the CIM_IndicationServiceCapabilities instance has a value of False, the implementation shall not allow the ModifyInstance operation to change the value of the SubscriptionRemovalTimeInterval property of the CIM_IndicationService instance.

8.4 CIM_IndicationServiceCapabilities

All operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

8.5 CIM_IndicationServiceSettingData

All operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

8.6 CIM_IndicationFilter

Table 4 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 4, all operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

Table 4 – Operations: CIM_IndicationFilter

Operation	Requirement	Messages
CreateInstance	Conditional	See 8.6.1.
DeleteInstance	Conditional	See 8.6.2.
ModifyInstance	Optional	See 8.6.3.

8.6.1 CIM_IndicationFilter — CreateInstance

This section details the requirements for the CreateInstance operation applied to an instance of CIM_IndicationFilter.

8.6.1.1 General Requirements

The implementation shall return a status code of CIM_ERROR_NOT_SUPPORTED in response to the CreateInstance method invoked by the client if the indication service is unable to support the indication filter. If an error is returned, the subscription is not activated.

If the CIM_IndicationFilter is valid and the indication service is able to support it, the implementation shall create an instance of CIM_ServiceAffectsElement that associates the CIM_IndicationFilter instance to the instance of CIM_IndicationService.

If a client attempts to create an instance of CIM_IndicationFilter by using the CreateInstance operation and the implementation determines that the query is invalid or not supportable, the implementation shall reject the operation and return a status code of CIM_ERROR_INVALID_PARAMETER in a CIM_Error instance response.

If a client attempts to create an instance of CIM_IndicationFilter by using the CreateInstance operation and dynamic filters are not supported by the implementation in this case, the implementation shall reject the operation and return a status code of CIM_ERROR_NOT_SUPPORTED in a CIM_Error instance response.

If a client attempts to create an instance of CIM_IndicationFilter by using the CreateInstance operation and the implementation is able to determine that an identical instance of CIM_IndicationFilter exists, the implementation should reject the operation and return a status code of CIM_ERROR_ALREADY_EXISTS in a CIM_Error instance response. The existing CIM_IndicationFilter instance object path shall be specified in the returned CIM_Error.ErrorSource instance property.

Clients should not populate the key properties of CIM_IndicationFilter when performing the CreateInstance operation. If the client populates the key properties of CIM_IndicationFilter, the implementation shall ignore these properties.

8.6.1.2 Conditional Requirement

The CreateInstance operation shall be supported for CIM_IndicationFilter if either of the following conditions is met:

- The CIM_IndicationService.FilterCreationEnabled property has the value True.
- An associated instance of CIM_IndicationServiceCapabilities exists, and the CIM_IndicationServiceCapabilities.FilterCreationEnabledIsSettable property has the value True.

8.6.2 CIM_IndicationFilter — DeleteInstance

This section details the requirements for the DeleteInstance operation applied to an instance of CIM_IndicationFilter.

8.6.2.1 General Requirements

If the instance of CIM_IndicationFilter is referenced by one or more instances of CIM_IndicationSubscription, the DeleteInstance operation shall not delete the CIM_IndicationFilter instance. If the CIM_IndicationFilter instance is not deleted, the operation shall return an error.

If an instance of CIM_IndicationFilter is deleted, all instances of CIM_ServiceAffectsElement that reference the instance of CIM_IndicationFilter shall also be deleted by the implementation.

If a client attempts to delete a static instance of CIM_IndicationFilter by using the DeleteInstance operation, the implementation shall reject the operation and return a status code of CIM_ERROR_NOT_SUPPORTED.

8.6.2.2 Conditional Requirement

The DeleteInstance operation shall be supported for CIM_IndicationFilter if either of the following conditions is met:

- The CIM_IndicationService.FilterCreationEnabled property has the value True.
- An associated instance of CIM_IndicationServiceCapabilities exists, and the CIM_IndicationServiceCapabilities.FilterCreationEnabledIsSettable property has the value True.

8.6.3 CIM_IndicationFilter — ModifyInstance

The ModifyInstance operation may be supported for an instance of CIM_IndicationFilter that represents a dynamic filter. The ModifyInstance operation may be supported for an instance of CIM_IndicationFilter that represents a static filter that is not defined by a profile. The ModifyInstance operation shall not be supported for an instance of CIM_IndicationFilter that represents a static filter defined by a profile.

8.7 CIM_FilterCollection

All operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

8.8 CIM_ListenerDestination

Table 5 lists implementation requirements for operations. If implemented, these operations shall be implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 5, all operations in the default list in 8.1 shall be implemented as defined in DSP0200.

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

1366 **Table 5 – Operations: CIM_ListenerDestination**

Operation	Requirement	Messages
CreateInstance	Optional	See 8.8.1.
DeleteInstance	Optional	See 8.8.2.
ModifyInstance	Optional	See 8.8.3.

1367 ### 8.8.1 CIM_ListenerDestination — CreateInstance

1368 This section details the requirements for the CreateInstance operation applied to an instance of
1369 CIM_ListenerDestination.

1370 Upon successful creation of the instance of CIM_ListenerDestination, the implementation shall create an
1371 instance of CIM_ServiceAffectsElement in which the AffectedElement property value references the
1372 instance of CIM_ListenerDestination created and the Service property references the instance of the
1373 CIM_IndicationService that can manage the listener destination information.

1374 If as many instances of CIM_ListenerDestination exist as the value of the
1375 CIM_IndicationServiceCapabilities.MaxListenerDestination property, the CreateInstance method shall fail.

1376 ### 8.8.2 CIM_ListenerDestination — DeleteInstance

1377 This section details the requirements for the DeleteInstance operation applied to an instance of
1378 CIM_ListenerDestination.

1379 If the instance of CIM_ListenerDestination is referenced by one or more instances of
1380 CIM_IndicationSubscription or CIM_FilterCollectionSubscription, the DeleteInstance operation shall not
1381 delete the CIM_ListenerDestination instance. Otherwise, if the CIM_ListenerDestination instance is not
1382 deleted, the operation shall return an error.

1383 When an instance of CIM_ListenerDestination is deleted, all instances of CIM_ServiceAffectsElement in
1384 which the AffectedElement property value references the instance of CIM_ListenerDestination to be
1385 deleted shall also be deleted.

1386 ### 8.8.3 CIM_ListenerDestination — ModifyInstance

1387 The ModifyInstance operation may be supported for an instance of CIM_ListenerDestination.

1388 ## 8.9 CIM_IndicationSubscription

1389 Table 6 lists implementation requirements for operations. If implemented, these operations shall be
1390 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 6, all operations in
1391 the default list in 8.1 shall be implemented as defined in DSP0200.

1392 NOTE: Related profiles may define additional requirements on operations for the profile class.

1393 **Table 6 – Operations: CIM_IndicationSubscription**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None
CreateInstance	Conditional. See 8.9.1.	None
DeleteInstance	Conditional. See 8.9.2.	None
ModifyInstance	Optional. See 8.9.3.	None

1394 ### 8.9.1 CIM_IndicationSubscription — CreateInstance

1395 This section details the requirements for the CreateInstance operation applied to an instance of
1396 CIM_IndicationSubscription.

1397 Support for the CreateInstance operation is conditional. The CreateInstance operation shall be supported
1398 if at least one instance of CIM_IndicationFilter is associated with the CIM_IndicationService through an
1399 instance of CIM_ServiceAffectsElement, where the CIM_IndicationFilter.IndividualSubscriptionSupported
1400 property has the value True.

1401 The CreateInstance operation shall return a status code of CIM_ERROR_NOT_SUPPORTED if the
1402 referenced instance of CIM_IndicationFilter is not valid. If an error is returned, the subscription is not
1403 activated. Successful creation of an instance of CIM_IndicationSubscription activates the client's
1404 subscription for delivery of the indications selected by the specified indication filter to the specified
1405 listener.

1406 The CreateInstance operation shall return a status code of CIM_ERROR_NOT_SUPPORTED if the value
1407 of the CIM_IndicationFilter.IndividualSubscriptionSupported property is False for the referenced instance
1408 of CIM_IndicationFilter.

1409 ### 8.9.2 CIM_IndicationSubscription — DeleteInstance

1410 This section details the requirements for the DeleteInstance operation applied to an instance of
1411 CIM_IndicationSubscription.

1412 Support for the DeleteInstance operation is conditional. The DeleteInstance operation shall be supported
1413 if at least one instance of CIM_IndicationFilter is associated with the CIM_IndicationService instance
1414 through an instance of CIM_ServiceAffectsElement, where the
1415 CIM_IndicationFilter.IndividualSubscriptionSupported property has the value True.

1416 Upon deletion of an instance of CIM_IndicationSubscription, the subscription is deactivated and the
1417 listener identified by that listener destination is no longer considered subscribed to that subscription.

1418 ### 8.9.3 CIM_IndicationSubscription — ModifyInstance

1419 The ModifyInstance operation may be supported for an instance of CIM_IndicationSubscription.

1420 ## 8.10 CIM_FilterCollectionSubscription

1421 Table 7 lists implementation requirements for operations. If implemented, these operations shall be
1422 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 7, all operations in
1423 the default list in 8.1 shall be implemented as defined in DSP0200.

1424 NOTE: Related profiles may define additional requirements on operations for the profile class.

1425 **Table 7 – Operations: CIM_FilterCollectionSubscription**

Operation	Requirement	Messages
Associators	Unspecified	None
AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None
CreateInstance	Mandatory. See 8.10.1.	None
DeleteInstance	Mandatory. See 8.10.2.	None
ModifyInstance	Optional. See 8.10.3.	None

1426 ### 8.10.1 CIM_FilterCollectionSubscription — CreateInstance

1427 This section details the requirements for the CreateInstance operation applied to an instance of
1428 CIM_FilterCollectionSubscription.

1429 Successful creation of an instance of CIM_FilterCollectionSubscription activates the client's subscription
1430 for delivery of the indications selected by the indication filters that are members of the collection
1431 subscribed to. Subscriptions are also recursively activated to collections that are members of the
1432 collection subscribed to.

1433 ### 8.10.2 CIM_FilterCollectionSubscription — DeleteInstance

1434 This section details the requirements for the DeleteInstance operation applied to an instance of
1435 CIM_FilterCollectionSubscription.

1436 When an instance of CIM_FilterCollectionSubscription is deleted, the subscription is deactivated and the
1437 listener identified by that listener destination is no longer considered subscribed to that subscription.

1438 ### 8.10.3 CIM_FilterCollectionSubscription — ModifyInstance

1439 The ModifyInstance operation may be supported for an instance of CIM_FilterCollectionSubscription.

1440 ## 8.11 CIM_ServiceAffectsElement

1441 Table 8 lists implementation requirements for operations. If implemented, these operations shall be
1442 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 8, all operations in
1443 the default list in 8.1 shall be implemented as defined in DSP0200.

1444 NOTE: Related profiles may define additional requirements on operations for the profile class.

1445 **Table 8 – Operations: CIM_ServiceAffectsElement**

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

1446 ## 8.12 CIM_MemberOfCollection

1447 Table 9 lists implementation requirements for operations. If implemented, these operations shall be
1448 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 9, all operations in
1449 the default list in 8.1 shall be implemented as defined in DSP0200.

1450 NOTE: Related profiles may define additional requirements on operations for the profile class.

1451 **Table 9 – Operations: CIM_MemberOfCollection**

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

1452 ## 8.13 CIM_ElementSettingData

1453 Table 10 lists implementation requirements for operations. If implemented, these operations shall be
1454 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 10, all operations
1455 in the default list in 8.1 shall be implemented as defined in DSP0200.

1456 NOTE: Related profiles may define additional requirements on operations for the profile class.

1457 **Table 10 – Operations: CIM_ElementSettingData**

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

1458 ## 8.14 CIM_OwningCollectionElement

1459 Table 11 lists implementation requirements for operations. If implemented, these operations shall be
1460 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 11, all operations
1461 in the default list in 8.1 shall be implemented as defined in DSP0200.

1462 NOTE: Related profiles may define additional requirements on operations for the profile class.

1463 **Table 11 – Operations: CIM_OwningCollectionElement**

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

1464 ## 8.15 CIM_ConcreteDependency

1465 Table 12 lists implementation requirements for operations. If implemented, these operations shall be
1466 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 12, all operations
1467 in the default list in 8.1 shall be implemented as defined in DSP0200.

1468 NOTE: Related profiles may define additional requirements on operations for the profile class.

1469 **Table 12 – Operations: CIM_ConcreteDependency**

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

1470 ## 8.16 CIM_HostedService

1471 Table 13 lists implementation requirements for operations. If implemented, these operations shall be
1472 implemented as defined in DSP0200. In addition, and unless otherwise stated in Table 13, all operations
1473 in the default list in 8.1 shall be implemented as defined in DSP0200.

1474 NOTE: Related profiles may define additional requirements on operations for the profile class.

1475 **Table 13 – Operations: CIM_HostedService**

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

9 Use Cases

This clause provides informative use cases and object diagrams.

9.1 Object Diagrams

For simplicity, the prefix *CIM_* has been removed from the names of the classes.

Figure 3 is an object diagram showing a possible implementation of the profile. In this diagram, the optional indications defined are supported. This support is indicated by the existence of fc2 associated through the CIM_ConcreteDependency instance with rp1. Mandatory indication filters and an optional vendor-defined collection of filters are defined for the *CPU Profile* as well. This is indicated by the existence of fc3 and fc4 associated with rp2 through the CIM_ConcreteDependency instance.

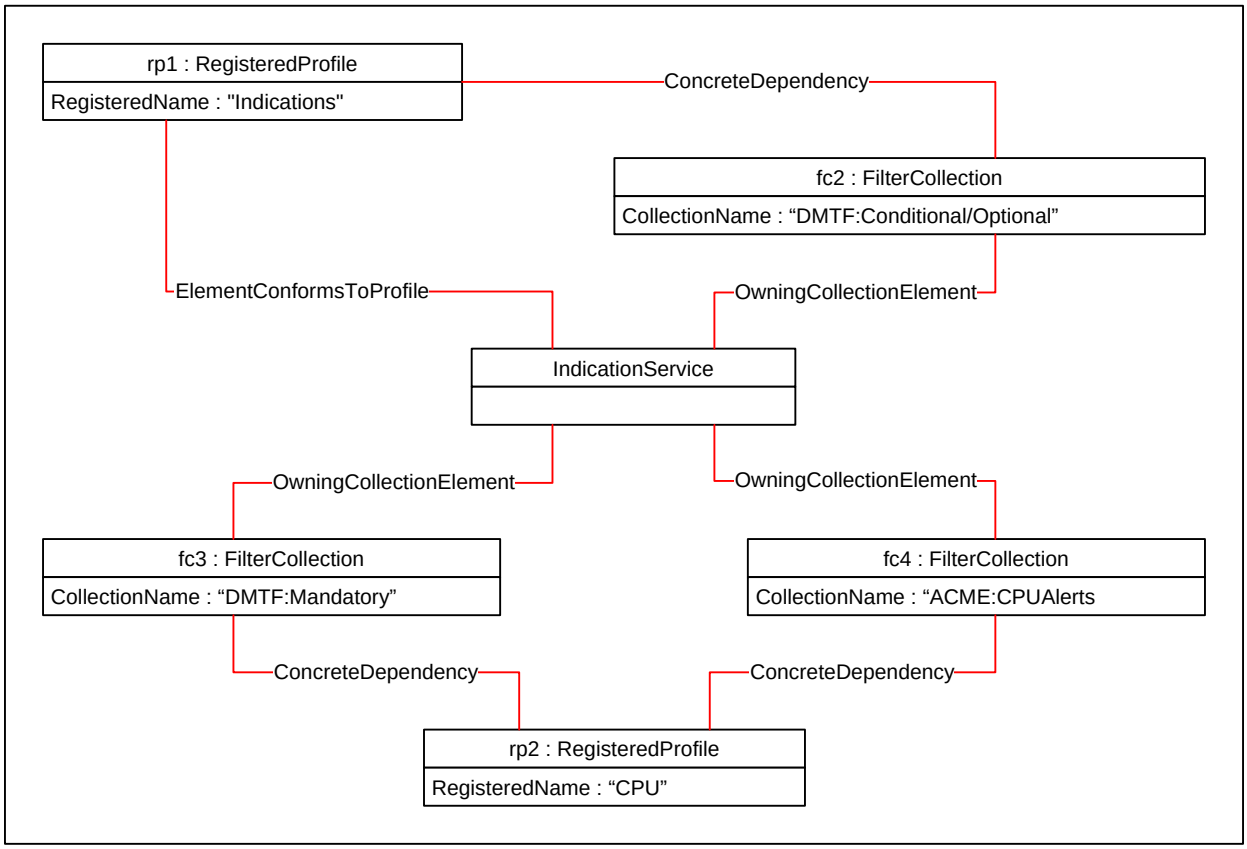
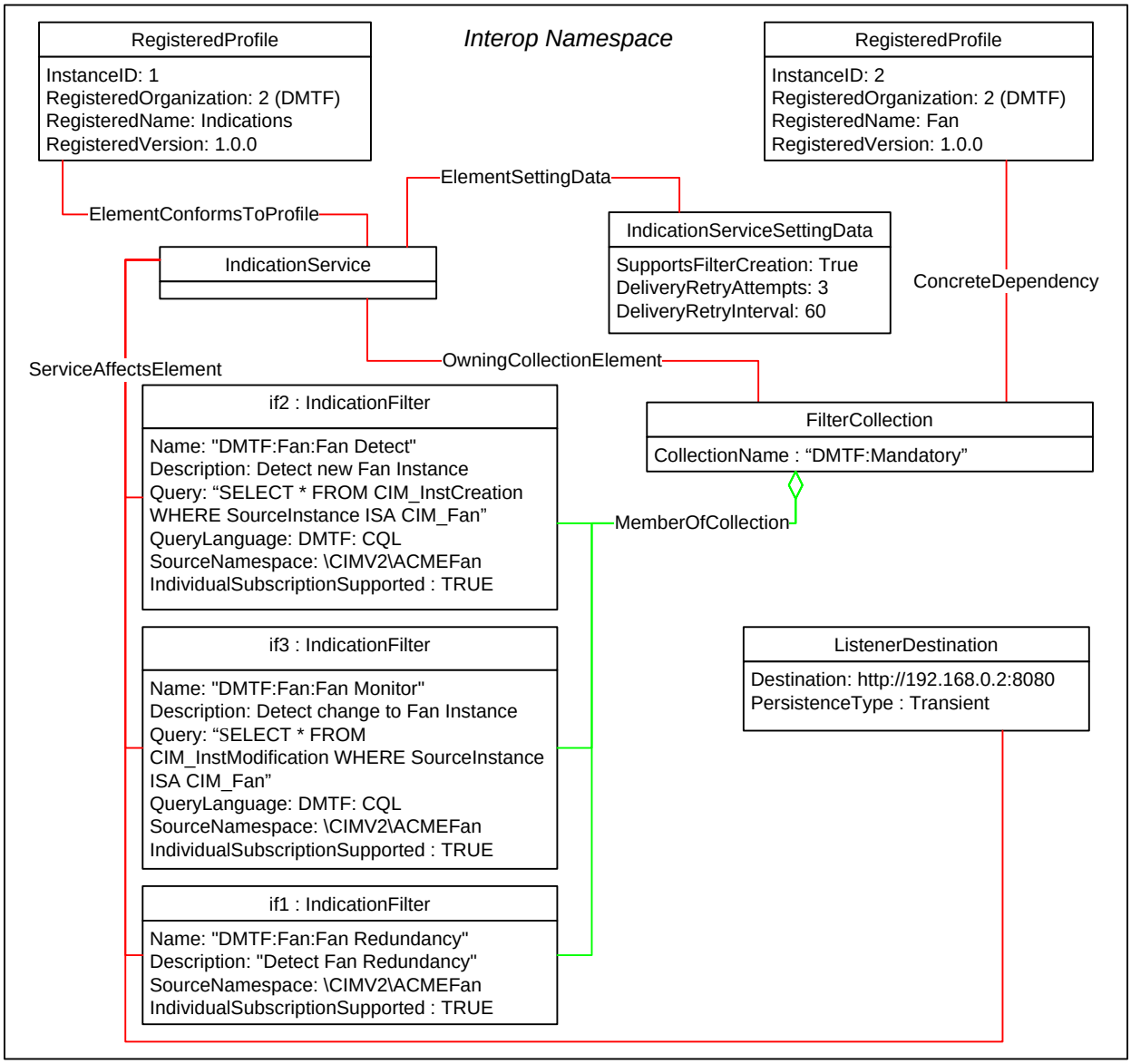


Figure 3 – Filter Collections Instance Diagram

1488 Figure 4 is an object diagram showing an implementation that supports mandatory indications defined in
1489 the *Fan Profile*. The implementation has explicitly instantiated instances of CIM_IndicationFilter to
1490 represent three of the mandatory indication filters. if2 and if3 are filters for lifecycle indications. if1 is a
1491 filter for alert indications related to changes in the status of fan redundancy.



1492

1493 **Figure 4 – Indications Profile Instance Diagram**

Figure 5 shows the same implementation as Figure 4 with the addition of individual subscriptions for each of the individually modeled indication filters. The three individual indication instances, ind1, ind2, and ind3, match these indication filters.

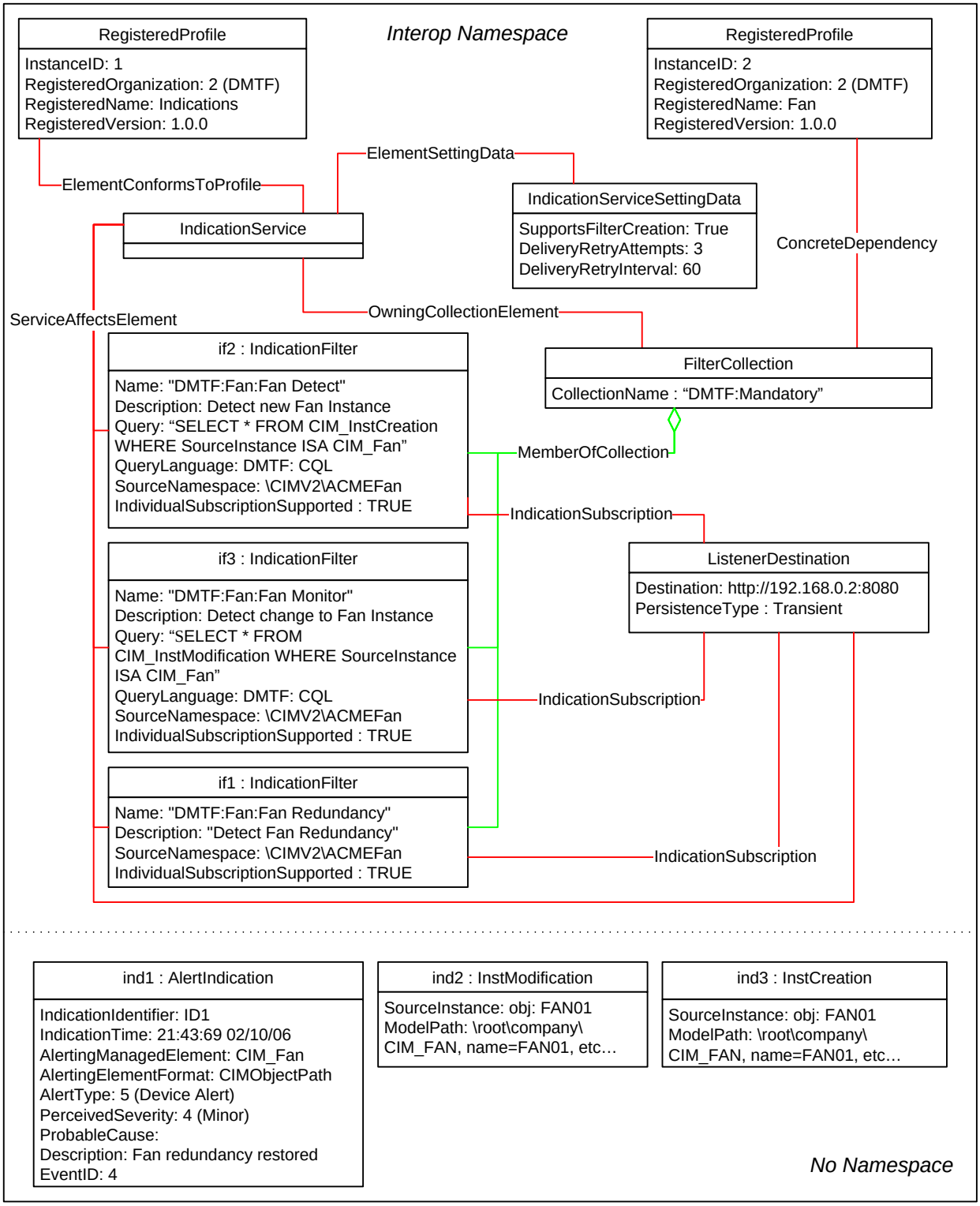
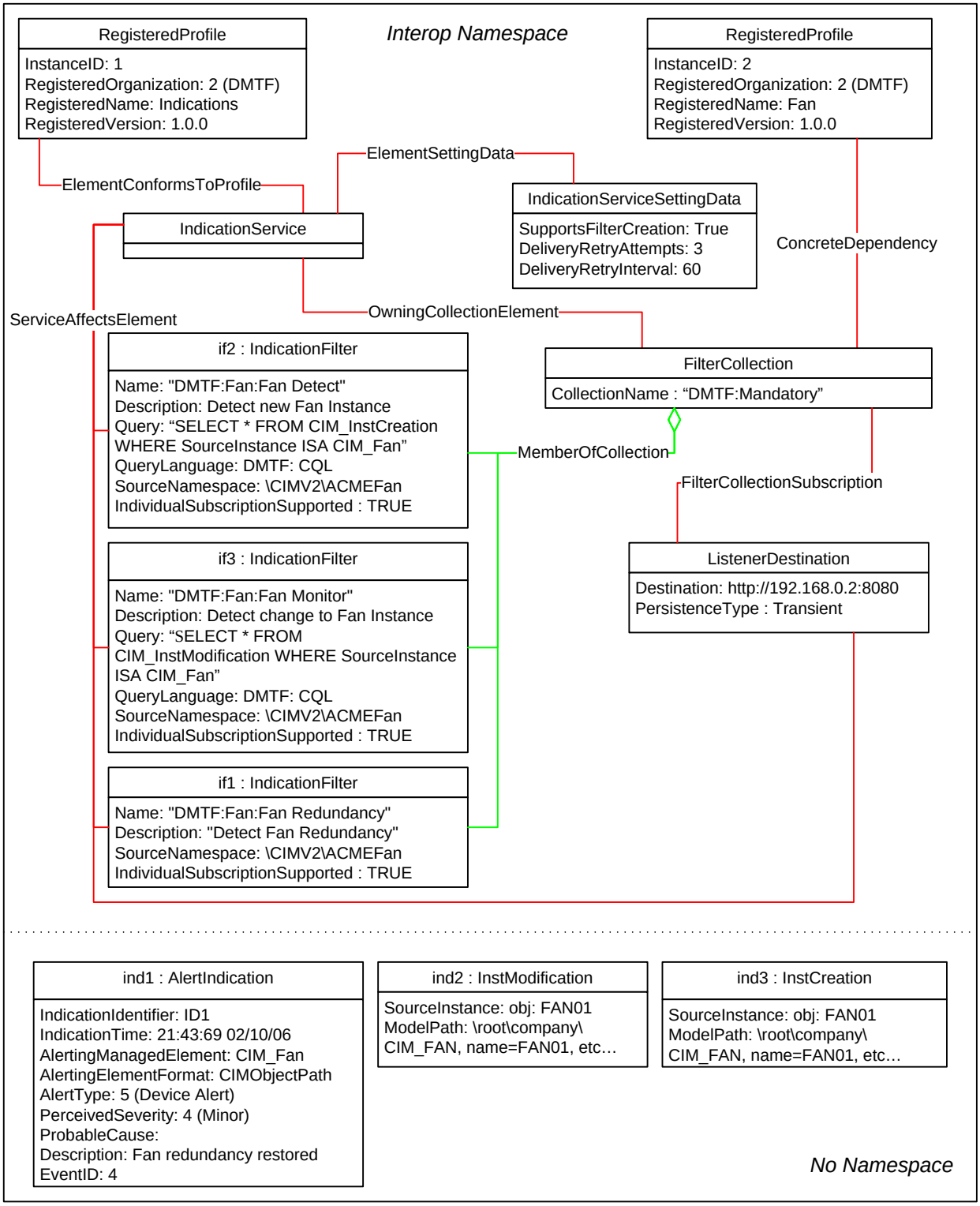


Figure 5 – Individual Subscriptions

1499 Figure 6 is an object diagram for the same implementation as Figure 4 with the addition of a collection
1500 subscription. The three individual indication instances, ind1, ind2, and ind3, match the indication filters
1501 contained in the CIM_FilterCollection instance.



1502

1503 **Figure 6 – Collection Subscription**

Figure 7 is an object diagram for the same implementation shown in Figure 4. A subscription has been created for the filter collection as well as an individual subscription to if1. This results in the duplicate notification ind1 and ind2.

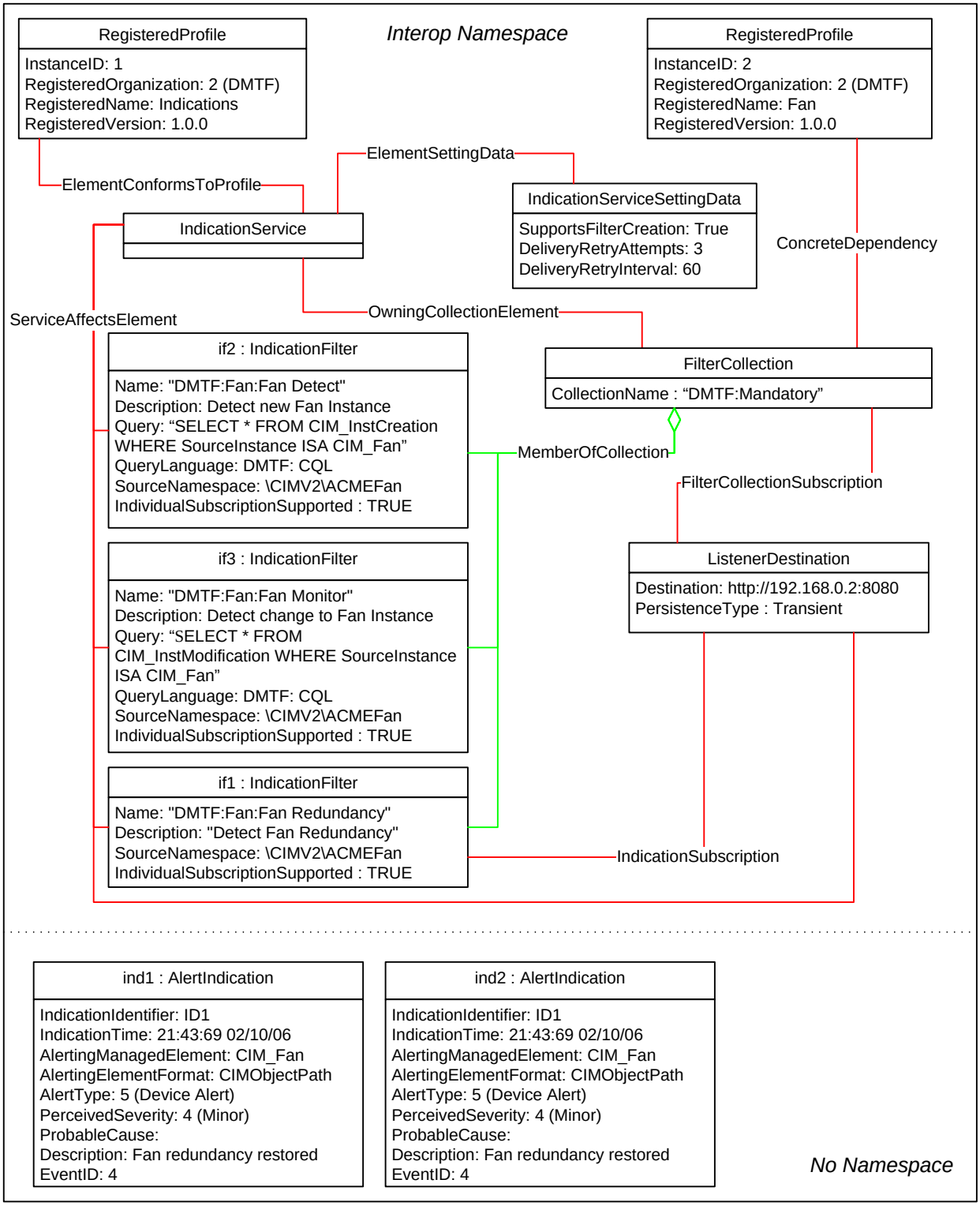


Figure 7 – Duplicate Subscriptions

Figure 8 is an object diagram for an implementation that supports a fixed number of listener destinations. A client selects one of the existing instances of CIM_ListenerDestination and modifies it appropriately to specify a desired destination for indication delivery. The implementation supports three listener destinations, which is indicated by the CIM_IndicationServiceCapabilities.MaxListenerDestinations property. The implementation statically creates instances of CIM_ListenerDestination. ld3 is currently configured to represent a transient listener destination. ld1 and ld2 are not configured and could be used by a client to identify desired destinations.

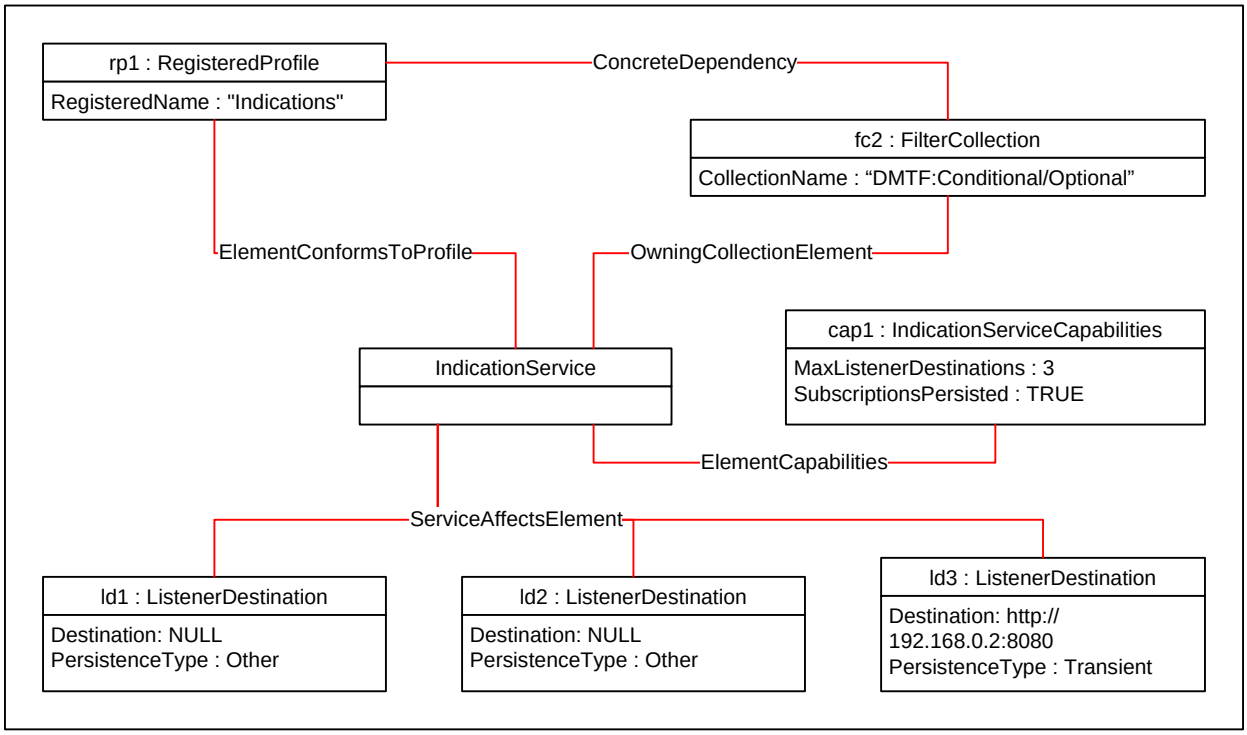


Figure 8 – Statically Provided Listener Destinations

9.2 Determine Whether Dynamic Filters Are Supported

Given an instance of CIM_IndicationService, a client can determine if dynamic filters are supported as follows:

1) Query the CIM_IndicationService.FilterCreationEnabled property. If the property has the value True, dynamic filters are supported.
2) If the property is False, find the associated instance of CIM_IndicationServiceCapabilities.
3) If an instance is found, query the value of the FilterCreationEnabledIsSettable property.
4) If FilterCreationEnabledIsSettable is True, modify the CIM_IndicationService, setting the FilterCreateEnabled property to True.
5) If the modification is successful, creating dynamic filters is supported. If the modification is unsuccessful, creating dynamic filters is not supported.

9.3 Create a Dynamic Filter for Alert Indications

Given the Owning Entity and Message Identifier for a standard message, a client can create a dynamic filter for an alert indication as follows:

1) Determine if dynamic filter creation is supported using the steps in 9.2.
2) If dynamic filter creation is supported, determine the query languages supported for indication filters using the steps in 9.17.
3) Using one of the supported query languages, create an instance of CIM_IndicationFilter in which the QueryLanguage property identifies one of the supported query languages and the Query property constrains the CIM_AlertIndication.OwningEntity and CIM_AlertIndication.MessageId properties to be the desired values.

9.4 Select a Listener Destination for Delivery of Indications

Given a listener to which the client wants to have indications delivered, a client can ensure that an appropriate CIM_ListenerDestination exists, as follows:

1) Find all instances of CIM_ListenerDestination that are associated with the CIM_IndicationService through an instance of CIM_ServiceAffectsElement.
2) For each instance of CIM_ListenerDestination, query the Destination property to determine if it represents the desired destination for indication delivery.

If an instance of CIM_ListenerDestination is not found, the client can use CreateInstance (or an equivalent operation) to create a new instance of CIM_ListenerDestination for indication delivery by specifying an appropriate instance of CIM_ListenerDestination as input to the operation.

9.5 Create a Subscription for a Single Filter

Given a desired destination for indication delivery and a desired filter, a client can create a subscription for an indication filter as follows:

1) Find all instances of CIM_IndicationFilter that are associated with the CIM_IndicationService instance through an instance of CIM_ServiceAffectsElement.
2) For each instance of CIM_IndicationFilter, evaluate the QueryLanguage and Query properties to determine if the CIM_IndicationFilter represents the desired indication filter.
3) If an instance of CIM_IndicationFilter is found, query the IndividualSubscriptionSupported property to determine if the implementation supports subscribing to this filter individually. If the property is True, individual subscription to this filter is supported. If the property is False, subscription to the individual filter is not supported and a dynamic filter needs to be created using the steps in 9.3.
4) Using the steps in 9.4, select an instance of CIM_ListenerDestination that represents the desired destination.
5) Use CreateInstance (or an equivalent) operation to create an instance of CIM_IndicationSubscription that references the CIM_IndicationFilter from step 3) and the CIM_ListenerDestination from step 4).

9.6 Subscribe for All Mandatory Indications for a Profile

A client can subscribe a listener for all of the mandatory indications defined for a profile as follows:

1) Determine if mandatory indications are supported for the profile.
2) If mandatory indications are supported for the profile, use the steps in 9.18 to subscribe to the CIM_FilterCollection instance that represents the mandatory filters.

9.7 Determine Whether a Subscription Exists for a Given Filter and Destination

A client can determine whether a subscription exists for a particular destination and filter as follows:

1) Find all instances of CIM_ListenerDestination that are associated with the CIM_IndicationService instance through an instance of CIM_ServiceAffectsElement.
2) For each instance of CIM_ListenerDestination, if the Destination property identifies the destination of interest, perform the following steps:
    a) Find all instances of CIM_IndicationFilter that are associated with the CIM_ListenerDestination instance through an instance of CIM_IndicationSubscription.
    b) For each instance of CIM_IndicationFilter, if the QueryLanguage and Query properties match the filter of interest, a subscription exists for the given filter and destination.
    c) Find all instances of CIM_FilterCollection that are associated with the CIM_ListenerDestination instance through an instance of CIM_IndicationFilterSubscription.
    d) For each instance of CIM_FilterCollection, evaluate the CIM_FilterCollection.CollectionName property to determine if the client has knowledge of filters contained in the collection.
3) If the client has knowledge, determine whether the CIM_FilterCollection instance contains the filter of interest. If it does, a subscription exists for the given filter and destination.
4) If the client does not have knowledge, find all instances of CIM_IndicationFilter that are associated with the CIM_FilterCollection instance through an instance of CIM_MemberOfCollection. For each instance of CIM_IndicationFilter, if the Query property matches the filter of interest, a subscription exists for the given filter and destination.

9.8 Determine the Components for Which Lifecycle Indications Are Available

Given an instance of CIM_IndicationFilter that filters for lifecycle indications, a client can determine the components for which the specified lifecycle indications can be provided, as follows:

1) Find the instances of CIM_FilterCollection with which the CIM_IndicationFilter instance is associated through an instance of CIM_MemberOfCollection.
    a) For each instance of CIM_FilterCollection, find the associated instances of CIM_RegisteredProfile.
    b) For each instance of CIM_RegisteredProfile, find the instances of CIM_ManagedElement that are in the scope of the profile.
    c) For each instance of CIM_ManagedElement, determine if it is implemented in a namespace identified by one of the values of the CIM_IndicationFilter.SourceNamespaces property, or if it is in the same namespace as the instance of CIM_IndicationFilter.
    d) For each instance of CIM_ManagedElement, determine if it matches the query specified by the QueryLanguage and Query properties of the CIM_IndicationFilter.

        If it matches the query, lifecycle indications filtered by the CIM_IndicationFilter are available for the CIM_ManagedElement instance.
2) If the instance of CIM_IndicationFilter is not associated with any instances of CIM_FilterCollection, determine the namespaces to which the filter applies by querying the value of the SourceNamespaces property.

    If the SourceNamespaces property is empty, the CIM_IndicationFilter applies to the namespace in which it is instantiated.

If the SourceNamespaces property is not empty, the CIM_IndicationFilter applies to each identified namespace.

3) For each instance of CIM_ManagedElement, determine if it matches the query specified by the Query property of the CIM_IndicationFilter. If it matches the query, lifecycle indications filtered by the CIM_IndicationFilter are available for the CIM_ManagedElement instance.

9.9 Subscribe for Indications of a Particular Severity

A client can subscribe a listener for all indications of a particular severity as follows:

Construct a query to select all instances of CIM_AlertIndication in which the PerceivedSeverity property has the desired value. Use this query as the input in the steps in 9.5.

9.10 Find the Scoping System for Which an Alert Indication Originated

Given an instance of CIM_AlertIndication, a client can determine the scoping system for which an indication originated, as follows:

1) Starting with the value of the CIM_AlertIndication.AlertingManagedElement property, retrieve the CIM element identified.

2) Using knowledge of profile definitions that contain the element, determine the profile with which the CIM element is conformant.

3) Use the algorithm defined for the profile to find the Scoping Instance.

9.11 Remove a Subscription

Given an instance of CIM_IndicationSubscription that represents an indication subscription, a client can remove the subscription as follows:

1) Invoke the DeleteInstance operation on the instance of CIM_IndicationSubscription.

2) If the previously referenced instance of CIM_IndicationFilter was a dynamic filter created by the client, no other instances of CIM_IndicationSubscription reference it, and the client does not plan to create a new subscription for this filter, the client can delete the CIM_IndicationFilter.

3) If the previously referenced instance of CIM_ListenerDestination was created by the client, no other instances of CIM_IndicationSubscription or CIM_FilterCollectionSubscription reference it, and the client does not plan to create a new subscription for this destination, the client can delete the CIM_ListenerDestination.

9.12 Remove a Listener Destination

A client can remove a listener destination as follows:

1) Remove each indication subscription configured for the destination by using the steps in 9.11.

2) Remove the listener destination by invoking the DeleteInstance operation on the instance of CIM_ListenerDestination.

9.13 Determine the Query That Triggered an Alert Indication

Given an instance of CIM_AlertIndication, a client can determine the indication filter that triggered an indication to be delivered, as follows:

1) Query the value of the CIM_AlertIndication.IndicationFilterName.

   If the value of the property identifies an indication filter of which the client has knowledge, the client knows the filter that caused the indication to be triggered.

2) If the value of the property does not identify an indication filter of which the client has knowledge, the client can find the indication filter as follows:

   a) Use the value of the CIM_AlertIndication.AlertingManagedElement property to find the implementation from which the indication originated.

   b) Find the instance of CIM_IndicationService in the Interop Namespace of the implementation.

   c) Find all instances of CIM_IndicationFilter that are associated with the CIM_IndicationService instance through an instance of CIM_ServiceAffectsElement.

   d) For each instance of CIM_IndicationFilter, determine if the value of the name property matches the value of the CIM_AlertIndication.IndicationFilterName property.

If it matches, the instance of CIM_IndicationFilter triggered the indication.

If a matching instance of CIM_IndicationFilter is not found, it is not possible for a client to determine the query.

Query the value of the CIM_IndicationFilter.Query and CIM_IndicationFilter.QueryLanguage properties to determine the query that resulted in the indication.

9.14 Configure the Number of Retries for Indication Delivery

A client can configure the number of retries attempted by an indication service as follows:

1) Find the instance of CIM_IndicationServiceCapabilities that is associated with the CIM_IndicationService instance through an instance of CIM_ElementCapabilities.

2) Query the value of the CIM_IndicationServiceCapabilities.DeliveryRetryAttemptsIsSettable property.

   a) If the value is True, use ModifyInstance to change the value of the CIM_IndicationService.DeliveryRetryAttempts to the desired value.

   b) If the value is False, the number of retries attempted by the CIM_IndicationService cannot be changed.

9.15 Modify a Dynamic Filter

A client can modify a dynamic filter as follows:

1) If the client maintained the object path of the instance of CIM_IndicationFilter that represents the dynamic filter, the client can invoke the DeleteInstance operation to remove the dynamic filter.

2) If the client has not maintained the object path, the client can find the dynamic filter to replace as follows:

   a) Find all instances of CIM_IndicationFilter that are associated with the CIM_IndicationService instance through an instance of CIM_ServiceAffectsElement.

   b) For each instance of CIM_IndicationFilter, determine if it matches the dynamic filter previously created.

   c) If it matches, attempt to modify the dynamic filter by using the ModifyInstance operation.

   d) If the ModifyInstance operation is not supported, invoke the DeleteInstance operation to remove it.

   e) Use the CreateInstance operation, specifying the desired attribute values, to create a new instance of CIM_IndicationFilter.

f) Replicate any CIM_IndicationSubscription instances that referenced the deleted instance of CIM_IndicationFilter, referencing the newly created CIM_IndicationFilter instance.

9.16 Filter for Indications from a Specific Namespace

A client can create a dynamic filter to receive indications from a specific namespace by using the steps in 9.3 with the additional constraint of specifying a value for the CIM_IndicationFilter.SourceNamespaces property.

9.17 Determine the Query Language Supported for Filtering Indications

A client can determine the query languages supported for filtering indications as follows:

1) Start with an empty set of supported query languages.

2) Find all instances of CIM_IndicationFilter that are associated with the CIM_IndicationService instance through an instance of CIM_ServiceAffectsElement.

3) For each instance of CIM_IndicationFilter, if the value of the CIM_IndicationFilter.QueryLanguage property is not included in the set from step 1), add it.

NOTE: The supported query languages can alternately be determined through knowledge of the implementation or through a combination of CIM elements and operations that are outside the scope of this profile.

9.18 Subscribe to All Events in a Collection

Given an instance of CIM_FilterCollection that represents a collection of indication filters and a desired destination for delivery of all indications in the collection, a client can create a subscription to all events in the collection as follows:

1) Select an instance of CIM_ListenerDestination that represents the desired destination by using the steps in 9.4.

2) Given the instance of CIM_ListenerDestination, create a subscription by creating an instance of CIM_FilterCollectionSubscription by using the CreateInstance operation (or equivalent), specifying the desired configuration of the subscription and references to the CIM_ListenerDestination instance and the CIM_FilterCollection instance.

9.19 Subscribe for All of the Indications Defined in a Profile

Given an instance of CIM_ListenerDestination that represents a desired destination for indication delivery, a client can subscribe a listener for all of the indications defined for implementations of a profile, as follows:

1) Enumerate instances of CIM_RegisteredProfile in the Interop namespace.

2) For each instance of CIM_RegisteredProfile, query the values of the RegisteredName, RegisteredVersion, and RegisteredOrganization properties to determine if the instance identifies the profile of interest.

3) If the instance of CIM_RegisteredProfile identifies the profile of interest:

   a) Find all instances of CIM_FilterCollection that are associated with the CIM_RegisteredProfile instance through an instance of CIM_ConcreteDependency.

      If no instances of CIM_FilterCollection are found, indications are not supported for the profile.

   b) For each instance of CIM_FilterCollection found, determine if it is referenced by an instance of CIM_MemberOfCollection, where it is the value of the Member reference.

1) If the CIM_FilterCollection instance is the value of the Member reference, find the CIM_FilterCollection instance that is the value of the Collection reference.

   - If the CIM_FilterCollection instance that is the value of the Collection reference is not associated with the CIM_RegisteredProfile instance from step 2), create an instance of CIM_FilterCollectionSubscription that references the CIM_FilterCollection instance that is the Member reference and the CIM_ListenerDestination instance that identifies the desired destination.
   - If the CIM_FilterCollection that is the value of the Collection reference is associated with the CIM_RegisteredProfile instance, skip it.

2) If the CIM_FilterCollection is not the value of the Member reference, create an instance of CIM_FilterCollectionSubscription that references the CIM_FilterCollection instance and the CIM_ListenerDestination instance that identifies the desired destination.

9.20 Determine the Maximum Number of Listener Destinations

Given an instance of CIM_IndicationService, a client can determine the maximum number of supported listener destinations as follows:

1) Find the associated instance of CIM_IndicationServiceCapabilities.
2) If an instance is found, query the value of the MaxListenerDestinations property.

   If an instance is not found, the maximum number of listener destinations is unknown.

10 CIM Elements

Table 14 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be implemented as described in Table 14. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose additional requirements on these elements.

Table 14 – CIM Elements: Indications Profile

Element Name	Requirement	Description
Classes		
CIM_AlertIndication	Optional	See 10.1.
CIM_ConcreteDependency	Conditional	See 10.2.
CIM_ElementCapabilities	Conditional	See 10.3.
CIM_ElementSettingData	Conditional	See 10.4.
CIM_FilterCollection	Optional	See 10.5.
CIM_FilterCollectionSubscription	Optional	See 10.6.
CIM_HostedService	Mandatory	See 10.7.
CIM_IndicationFilter	Optional	See 10.8.
CIM_IndicationService	Mandatory	See 10.9.
CIM_IndicationServiceCapabilities	Optional	See 7.14 and 10.10.
CIM_IndicationServiceSettingData	Optional	See 7.2 and 10.11.
CIM_IndicationSubscription	Conditional	See 10.12.
CIM_InstCreation	Optional	See 10.13.
CIM_InstDeletion	Optional	See 10.14.

Element Name	Requirement	Description
CIM_InstModification	Optional	See 10.15.
CIM_ListenerDestination	Mandatory	See 10.16.
CIM_MemberOfCollection	Optional	See 10.17.
CIM_OwningCollectionElement	Conditional	See 10.18.
CIM_RegisteredProfile	Mandatory	See 10.19.
CIM_ServiceAffectsElement	Conditional	See 10.20.
Indications		
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_IndicationSubscription	Optional	See 7.16.2.3.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_FilterCollectionSubscription	Optional	See 7.16.2.4.
SELECT * FROM CIM_InstDeletion WHERE SourceInstance ISA CIM_ListenerDestination	Optional	See 7.16.2.2.

10.1 CIM_AlertIndication

CIM_AlertIndication is a specialized type of CIM_Indication that contains information about the severity, cause, recommended actions, and other data of a real world event. Profiles that define support for asynchronous notification of events can constrain this class and may require it. Table 15 contains the requirements for elements of this class.

Table 15 – Class: CIM_AlertIndication

Elements	Requirement	Notes
IndicationIdentifier	Mandatory	An identifier for the indication used for correlated indications
IndicationTime	Mandatory	The time and date of creation of the indication. The property may be set to NULL if it cannot be determined.
AlertingManagedElement	Mandatory	The identifying information for the element that changed, as a WBEM-URI-TypedInstancePath (as defined in DSP0207), of the entity for which this Indication is generated
AlertingElementFormat	Mandatory	Matches 2 (CIMObjectPath)
IndicationFilterName	Mandatory	See 7.15.
AlertType	Mandatory	Primary classification of the indication. This value depends on the content of the alert message and typically should be 5 (Device Alert) or 6 (Environmental Alert) for most hardware-related indications.
PerceivedSeverity	Mandatory	Describes the severity of the alert indication
ProbableCause	Mandatory	None
SystemName	Mandatory	Should be the value of the Name property of the scoping system of the managed element that is the AlertingManagedElement

Elements	Requirement	Notes
CorrelatedIndications	Optional	IndicationIdentifiers whose notifications are correlated with this one
OtherAlertType	Conditional	If AlertType matches 1 (Other), this property is mandatory. Pattern ("+.")
OtherSeverity	Conditional	If PerceivedSeverity matches 1 (Other), this property is mandatory.
ProbableCauseDescription	Conditional	If ProbableCause matches 1 (Other), this property is mandatory.
OwningEntity	Mandatory	See 7.11.
MessageID	Mandatory	See 7.11.
MessageArguments	Mandatory	See 7.11.
Message	Optional	See 7.11.

10.2 CIM_ConcreteDependency

CIM_ConcreteDependency is used to associate instances of CIM_FilterCollection to instances of CIM_RegisteredProfile. This association identifies the profile that provides context and scope to a collection of indication filters. The existence of instances of CIM_ConcreteDependency is conditional on the existence of instances of CIM_FilterCollection. Table 16 contains the requirements for elements of this class.

Table 16 – Class: CIM_ConcreteDependency

Elements	Requirement	Notes
Antecedent	Mandatory	Key: Shall reference the instance of CIM_RegisteredProfile that represents the profile for which the set of indications is supported Cardinality 1
Dependent	Mandatory	Key: Shall reference the instance of CIM_FilterCollection that represents the set of indications supported for this profile Cardinality *

10.3 CIM_ElementCapabilities

CIM_ElementCapabilities is used to associate an instance of CIM_IndicationServiceCapabilities with an instance of CIM_IndicationService. An instance of CIM_ElementCapabilities is conditional on the existence of an instance of CIM_IndicationServiceCapabilities. Table 17 contains the requirements for elements of this class.

1776 **Table 17 – Class: CIM_ElementCapabilities**

Elements	Requirement	Notes
ManagedElement	Mandatory	Key: Shall reference the Central Instance Cardinality 1
Capabilities	Mandatory	Key: Shall reference the instance of CIM_IndicationServiceCapabilities that represents the indication service property setting capabilities Cardinality 0..1

1777 ## 10.4 CIM_ElementSettingData

1778 CIM_ElementSettingData is used to associate an instance of CIM_IndicationServiceSettingData with an
1779 instance of CIM_IndicationService. An instance of CIM_ElementSettingData is conditional on the
1780 existence of an instance of CIM_IndicationServiceSettingData. Table 18 contains the requirements for
1781 elements of this class.

1782 **Table 18 – Class: CIM_ElementSettingData**

Elements	Requirement	Notes
ManagedElement	Mandatory	Key: Shall reference the instance of CIM_IndicationService that represents the implementation's support for indications Cardinality 1
SettingData	Mandatory	Key: Shall reference the instance of CIM_IndicationServiceSettingData that represents the indication service settings Cardinality 0..1
IsDefault	Mandatory	Matches 1 (Is Default)
IsNext	Mandatory	Matches 1 (Is Next)

1783 ## 10.5 CIM_FilterCollection

1784 CIM_FilterCollection represents collections of indication filters. Table 19 contains the requirements for
1785 elements of this class.

1786 **Table 19 – Class: CIM_FilterCollection**

Elements	Requirement	Notes
InstanceID	Mandatory	Key: Shall specify the unique identifier for an instance of this class within the Implementation namespace
CollectionName	Mandatory	See 7.6.4.

1787 ## 10.6 CIM_FilterCollectionSubscription

1788 CIM_FilterCollectionSubscription is used to associate an instance of CIM_FilterCollection with an instance
1789 of CIM_ListenerDestination. The existence of an instance of this class reflects the subscription to a
1790 collection of instances of CIM_IndicationFilter. The association shall imply a subscription to all the
1791 instances of CIM_IndicationFilter that are members of the collection. Support for this class is conditional
1792 on support for CIM_FilterCollection. Table 20 contains the requirements for elements of this class.

1793 **Table 20 – Class: CIM_FilterCollectionSubscription**

Elements	Requirement	Notes
Filter	Mandatory	Key: Shall reference the instance of CIM_FilterCollection that represents the set of indications to which a listener has been subscribed Cardinality *
Handler	Mandatory	Key: Shall reference the CIM_ListenerDestination that represents the location to which indications shall be delivered when they occur Cardinality *
OnFatalErrorPolicy	Mandatory	See 7.8.
OtherOnFatalErrorPolicy	Conditional	Mandatory if the value of OnFatalErrorPolicy is 1 (Other) Pattern (".+")
FailureTriggerTimeInterval	Mandatory	Specifies minimum delay before OnFatalErrorPolicy is implemented
SubscriptionState	Mandatory	None
OtherSubscriptionState	Conditional	Mandatory if the value of SubscriptionState is 1 (Other) Pattern (".+")
RepeatNotificationPolicy	Mandatory	Matches 2 (None), 3 (Suppress), or 4 (Delay)
RepeatNotificationInterval	Conditional	Mandatory if the value of RepeatNotificationPolicy is 3 (Suppress) or 4 (Delay)
RepeatNotificationGap	Conditional	Mandatory if the value of RepeatNotificationPolicy is 4 (Delay)
RepeatNotificationCount	Conditional	Mandatory if the value of RepeatNotificationPolicy is 3 (Suppress) or 4 (Delay)

1794 ## 10.7 CIM_HostedService

1795 CIM_HostedService is used to relate the CIM_IndicationService instance to its scoping CIM_System
1796 instance. Table 21 contains the requirements for elements of this class.

1797 **Table 21 – Class: CIM_HostedService**

Elements	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to the Scoping Instance. Cardinality 1
Dependent	Mandatory	This property shall be a reference to the Central Instance. Cardinality 1..*

10.8 CIM_IndicationFilter

CIM_IndicationFilter represents static and dynamic indication filters. CIM_IndicationFilter is optional. It is expected that referencing profiles define mandatory instances of CIM_IndicationFilter such that the class is further constrained to be mandatory in the referencing profile. Table 22 contains the requirements for elements of this class.

Table 22 – Class: CIM_IndicationFilter

Elements	Requirement	Notes
SystemCreationClassName	Mandatory	Key: Shall be populated by the implementation with the class name of the scoping system. If a value is supplied by the client, it shall be ignored by the implementation.
CreationClassName	Mandatory	Key: Shall be populated by the implementation with the name of the class of which this is an instance. If a value is supplied by the client, it shall be ignored by the implementation.
SystemName	Mandatory	Key: Shall be populated by the implementation with the name of the scoping system. If a value is supplied by the client, it shall be ignored by the implementation.
Name	Mandatory	Key: Shall be populated by the implementation with the unique name of the instance or as specified by profile-defined static filters or by the client when creating dynamic filters. See 7.4.8.
Query	Mandatory	Specifies the query that defines the filter. See 7.4.6.
QueryLanguage	Mandatory	Specifies the query language used for the filter. See 7.4.6.
SourceNamespaces	Mandatory	Specifies the source namespaces from which indications originate. See 7.4.7.
ElementName	Optional	A user-friendly string that describes the indication. Modification of this property by the client may or may not be supported.
IndividualSubscriptionSupported	Mandatory	None

10.9 CIM_IndicationService

CIM_IndicationService represents an indication service, which is a component in an implementation that performs the delivery of indications to a listener. This class is the Central Class of the profile. Table 23 contains the requirements for elements of this class.

Table 23 – Class: CIM_IndicationService

Elements	Requirement	Notes
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
Name	Mandatory	Key
FilterCreationEnabled	Mandatory	See 7.1.
DeliveryRetryAttempts	Mandatory	See 7.1.

Elements	Requirement	Notes
DeliveryRetryInterval	Mandatory	See 7.1.
SubscriptionRemovalAction	Mandatory	See 7.1.
SubscriptionRemovalTimeInterval	Mandatory	See 7.1.

10.10 CIM_IndicationServiceCapabilities

CIM_IndicationServiceCapabilities is an optional element that represents the capabilities of the CIM_IndicationService instance. Table 24 contains the requirements for elements of this class.

Table 24 – Class: CIM_IndicationServiceCapabilities

Element	Requirement	Notes
InstanceID	Mandatory	Key: Shall specify the unique identifier for an instance of this class within the Implementation namespace
FilterCreationEnabledIsSettable	Mandatory	Defines whether the client can modify the FilterCreationEnabled property of the associated CIM_IndicationService instance
DeliveryRetryAttemptsIsSettable	Mandatory	Defines whether the client can modify the DeliveryRetryAttempts property of the associated CIM_IndicationService instance
DeliveryRetryIntervalIsSettable	Mandatory	Defines whether the client can modify the DeliveryRetryInterval property of the associated CIM_IndicationService instance
SubscriptionRemovalActionIsSettable	Mandatory	Defines whether the client can modify the SubscriptionRemovalAction property of the associated CIM_IndicationService instance
SubscriptionRemovalTimeIntervalIsSettable	Mandatory	Defines whether the client can modify the SubscriptionRemovalTimeInterval property of the associated CIM_IndicationService instance
MaxListenerDestinations	Mandatory	Indicates the maximum number of listener destinations
MaxActiveSubscriptions	Mandatory	Indicates the maximum number of active subscriptions
SubscriptionsPersisted	Mandatory	Indicates whether subscriptions are persisted across restarts of the indication service

10.11 CIM_IndicationServiceSettingData

CIM_IndicationServiceSettingData is used to represent the initial configuration of the CIM_IndicationService instance. Table 25 contains the requirements for elements of this class.

1816 **Table 25 – Class: CIM_IndicationServiceSettingData**

Elements	Requirement	Notes
InstanceID	Mandatory	Key
FilterCreationEnabled	Mandatory	See 7.1.2.
DeliveryRetryAttempts	Mandatory	See 7.1.2.
DeliveryRetryInterval	Mandatory	See 7.1.2.
SubscriptionRemovalAction	Mandatory	See 7.1.2.
SubscriptionRemovalTimeInterval	Mandatory	See 7.1.2.

1817 10.12 CIM_IndicationSubscription

1818 CIM_IndicationSubscription is used to associate an instance of CIM_IndicationFilter with an instance of
1819 CIM_ListenerDestination. The existence of an instance of this class reflects the subscription to a single
1820 CIM_IndicationFilter instance. CIM_IndicationSubscription is conditional. Instances of
1821 CIM_IndicationSubscription may exist if at least one instance of CIM_IndicationFilter is associated with
1822 the Central Instance through an instance of CIM_ServiceAffectsElement. Table 26 contains the
1823 requirements for elements of this class.

1824 **Table 26 – Class: CIM_IndicationSubscription**

Elements	Requirement	Notes
Filter	Mandatory	Key: Shall reference the instance of CIM_IndicationFilter that represents the indication to which a listener has been subscribed
Handler	Mandatory	Key: Shall reference the CIM_ListenerDestination that represents the location to which the indication shall be delivered when it occurs
OnFatalErrorPolicy	Mandatory	None
OtherOnFatalErrorPolicy	Conditional	Mandatory if the value of OnFatalErrorPolicy is 1 (Other) Pattern (".+")
FailureTriggerTimeInterval	Mandatory	Specifies the minimum delay before OnFatalErrorPolicy is implemented
SubscriptionState	Mandatory	None
OtherSubscriptionState	Conditional	Mandatory if the value of SubscriptionState is 1 (Other) Pattern (".+")
RepeatNotificationPolicy	Mandatory	Matches 2 (None), 3 (Suppress), or 4 (Delay)
RepeatNotificationInterval	Conditional	Mandatory if the value of RepeatNotificationPolicy is 4 (Delay)
RepeatNotificationGap	Conditional	Mandatory if the value of RepeatNotificationPolicy is 3 (Suppress) or 4 (Delay)
RepeatNotificationCount	Conditional	Mandatory if the value of RepeatNotificationPolicy is 3 (Suppress) or 4 (Delay)

10.13 CIM_InstCreation

CIM_InstCreation notifies a handler when a new instance of a class is created. Referencing profiles that require asynchronous notification of instance creation use this class. Table 27 contains the requirements for elements of this class.

Table 27 – Class: CIM_InstCreation

Elements	Requirement	Notes
IndicationIdentifier	Mandatory	An identifier for the indication used for correlated indications. The value for this property should be unique for an extended period of time.
IndicationTime	Mandatory	The time and date of creation of the indication. This property shall be populated with a valid datetime value.
SourceInstance	Mandatory	A copy of the instance that changed to generate the indication. SourceInstance contains the current values of the properties selected by the Indication Filter's Query.
SourceInstanceModelPath	Mandatory	The identifying information, as a WBEM-URI-TypedInstancePath (as defined in DSP0207), of the entity for which this Indication is generated
IndicationFilterName	Mandatory	See 7.15.
CorrelatedIndications	Optional	IndicationIdentifiers whose notifications are correlated with this one

10.14 CIM_InstDeletion

CIM_InstDeletion notifies a handler when an instance of a class is deleted. Referencing profiles that require asynchronous notification of instance deletion use this class. Table 28 contains the requirements for elements of this class.

Table 28 – Class: CIM_InstDeletion

Elements	Requirement	Notes
IndicationIdentifier	Mandatory	An identifier for the indication used for correlated indications. The value for this property should be unique for an extended period of time.
IndicationTime	Mandatory	The time and date of creation of the indication. The property shall be populated with a valid datetime value.
SourceInstance	Mandatory	A copy of the instance that changed to generate the indication. SourceInstance contains the current values of the properties selected by the Indication Filter's Query.
SourceInstanceModelPath	Mandatory	The identifying information, as a WBEM-URI-TypedInstancePath (as defined in DSP0207), of the entity for which this Indication is generated
IndicationFilterName	Mandatory	See 7.15.
CorrelatedIndications	Optional	IndicationIdentifiers whose notifications are correlated with this one

10.15 CIM_InstModification

CIM_InstModification notifies a handler when an instance (of a class defined in the Filter QueryString) is modified or changed. Referencing profiles that require asynchronous notification of instance modification use this class. Table 29 contains the requirements for elements of this class.

Table 29 – Class: CIM_InstModification

Elements	Requirement	Notes
IndicationIdentifier	Mandatory	An identifier for the indication used for correlated indications. The value for this property should be unique for an extended period of time.
IndicationTime	Mandatory	The time and date of creation of the indication. The property shall be set with a valid datetime value.
SourceInstance	Mandatory	A copy of the instance that changed to generate the indication. SourceInstance contains the current values of the properties selected by the Indication Filter's Query.
SourceInstanceModelPath	Mandatory	The identifying information, as a WBEM-URI-TypedInstancePath (as defined in DSP0207), of the entity for which this Indication is generated
IndicationFilterName	Mandatory	See 7.15.
CorrelatedIndications	Optional	IndicationIdentifiers whose notifications are correlated with this one
PreviousInstance	Optional	A copy of the “previous” instance whose change generated the indication. PreviousInstance contains “older” values of an instance's properties (as compared to SourceInstance), selected by the Indication Filter's Query.

10.16 CIM_ListenerDestination

CIM_ListenerDestination represents a destination for the delivery of indications. Table 30 contains the requirements for elements of this class.

Table 30 – Class: CIM_ListenerDestination

Elements	Requirement	Notes
SystemCreationClassName	Mandatory	Key: Shall be populated by the implementation with the class name of the scoping system. If the client supplies a value, the implementation shall ignore it.
SystemName	Mandatory	Key: Shall be populated by the implementation with the name of the scoping system. If the client supplies a value, the implementation shall ignore it.
CreationClassName	Mandatory	Key: Shall be populated by the implementation with the name of the class of which this is an instance. If the client supplies a value, the implementation shall ignore it.
Name	Mandatory	Key: Shall be populated by the implementation with the unique name of the instance. If the client supplies a value, the implementation shall ignore it
PersistenceType	Mandatory	See 7.5.3.

Elements	Requirement	Notes
ElementName	Mandatory	A user-friendly string that describes the destination. Modification of this property by the client may or may not be supported.
Destination	Mandatory	See 7.5.2.
ProtocolType	Mandatory	Shall be specified by the client as one of the enumerations from the class definition

10.17 CIM_MemberOfCollection

CIM_MemberOfCollection is used to aggregate instances of CIM_IndicationFilter or instances of CIM_FilterCollection to an instance of CIM_FilterCollection. This class identifies an indication or collection of indications as being part of a specific collection of indications. Table 31 contains the requirements for elements of this class.

Table 31 – Class: CIM_MemberOfCollection

Elements	Requirement	Notes
Collection	Mandatory	Key: Shall reference an instance of CIM_FilterCollection Cardinality *
Member	Mandatory	Key: Shall reference an instance of CIM_IndicationFilter or CIM_FilterCollection Cardinality *

10.18 CIM_OwningCollectionElement

CIM_OwningCollectionElement is used to associate instances of CIM_FilterCollection with an instance of CIM_IndicationService. The existence of an instance of CIM_OwningCollectionElement is conditional on the existence of an instance of CIM_FilterCollection. Table 32 contains the requirements for elements of this class.

Table 32 – Class: CIM_OwningCollectionElement

Elements	Requirement	Notes
OwningElement	Mandatory	Key: Shall reference the Central Instance Cardinality 1
OwnedElement	Mandatory	Key: Shall reference an instance of CIM_FilterCollection Cardinality *

10.19 CIM_RegisteredProfile

CIM_RegisteredProfile identifies the *Indications Profile* in order for a client to determine whether support for indications is supported by the managed system instrumentation. The CIM_RegisteredProfile class is defined by the *Profile Registration Profile*. With the exception of the mandatory values specified for the elements in Table 33, the behavior of the RegisteredProfile instance is in accordance with the *Profile Registration Profile*.

1862 **Table 33 – Class: CIM_RegisteredProfile**

Elements	Requirement	Notes
RegisteredName	Mandatory	This property shall have a value of "Indications".
RegisteredVersion	Mandatory	This property shall have a value of "1.1.0".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

1863 ## 10.20 CIM_ServiceAffectsElement

1864 CIM_ServiceAffectsElement is used to associate instances of CIM_IndicationFilter and
1865 CIM_ListenerDestination with an instance of CIM_IndicationService. The existence of
1866 CIM_ServiceAffectsElement is conditional on the existence of at least one instance of
1867 CIM_IndicationFilter, CIM_ListenerDestination, or CIM_FilterCollection. Table 34 contains the
1868 requirements for elements of this class.

1869 **Table 34 – Class: CIM_ServiceAffectsElement**

Elements	Requirement	Notes
AffectingElement	Mandatory	Key: Shall reference the Central Instance Cardinality 1
AffectedElement	Mandatory	Key: Shall be a reference to an instance of CIM_IndicationFilter or CIM_ListenerDestination Cardinality *

1870

ANNEX A
(informative)

Profiles That Define Indications

Profiles that define indications document support in the following ways:

- Profiles shall define supported events in terms of lifecycle and alert indications within the “CIM Elements” table of the profile specification.
- A row included in the “Referenced Profiles” table of the “Synopsis” clause that specifies the *Indications Profile*. The “Relationship” column in the table contains *Mandatory* if mandatory indications are specified in the profile being defined.
- Normative text provided in the “Implementation” clause of the profile being defined, listing the indications being specified in the profile and in what circumstances they can be produced.
- The “CIM Elements” table in the “CIM Elements” clause of the profile being defined contains an entry for each indication being specified. The entry consists of the query for the indication; whether it is mandatory, conditional, or optional; and a description of the indication. Additionally, if a profile requires an instance of CIM_IndicationFilter to be instantiated to represent the indication, a subclause in Clause 7, "Implementation", is needed to make this normative requirement.
- CIM_IndicationFilter listed as a mandatory, conditional, or optional class within the profile based on requirements for static filters. Further each profile specifies, per indication definition, whether it is required that an implementation instantiate an instance of CIM_IndicationFilter for each indication definition.
- CIM_FilterCollection listed as a mandatory, conditional, or optional class within the profile based on profile requirements.

NOTE: The requirements for backwards compatibility when applied to the specification of indication filters in a profile are such that once an indication filter has been defined in a profile, all subsequent minor versions of the profile continue to specify the indication filter, while a subsequent major version may remove the requirement.

ANNEX B
(informative)

Change Log

Version	Date	Description
1.0.0a	2007-06-04	Released as Preliminary Standard
1.0.0	2008-12-05	Released as Final Standard
1.0.1	2009-09-07	Released as DMTF Standard, with the following changes: • Updated profile conventions for operations and their usage • Fixed incorrect CIM Schema version (from 2.16 to 2.22)
1.1.0a	2009-12-02	Released as Work in Progress, with the following changes: • Increased CIM Schema version to 2.24. • Added support for reliable indications (delivery retry, detection of lost indications, reconstruction of original order): – Description of reliable indications concept in 7.10 (Indication Delivery). – Clarifications in description of CIM_ListenerDestination.PersistenceType. • Refined the format for CIM_FilterCollection.CollectionName in 7.6. • Refined the format for CIM_IndicationFilter.Name in 7.4. • Cleaned up terminology clause by removing most terms that are defined in DSP0004, DSP0200 or DSP1001. • Added document conventions clause and consolidated existing text into that. • Updated profile conventions for operations to match DSP1001 1.0.1. • Fixed incorrect pattern value "WBEMURI" for CIM_AlertIndication.AlertingElementFormat.
1.1.0	2010-05-20	Released as DMTF Standard, with the following changes: • Clarified and added some terms in clause 3. • Clarified that there is only one indication service in a WBEM server, but added a recommendation for clients to expect more than one in the future. • Fixed incorrect verbiage of sending indications to clients, to sending indications to listeners. • Changed ambiguous "conditional/optional" requirement to "conditional or optional" in all cases but one. • Clarified that listeners that intend to re-establish the original order of indications need to buffer indications that do not have the predicted sequence number until decision about loss can be made. • Lowered the requirement not to interpret sequence numbers in case of not implementing them, to a permission to ignore them. • Fixed inconsistencies in several diagrams.