Document Number: DSP1003

Date: 2007-02-12

Version: 1.0.0a

Policy Profile

Document Type: Specification

Document Status: Preliminary Standard

Document Language: E

CONTENTS

Figures

Tables

Foreword

The *Policy Profile* (DSP1003) was prepared by the Policy Working Group of the DMTF.

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

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 and manipulated to represent and manage policy collections and rules modeled using the DMTF Common Information Model (CIM) core and extended model definitions.

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

177 # Policy Profile

178 ## 1 Scope

179 The *Policy Profile* is an abstract profile that extends the management capability of referencing profiles by
180 adding the capability to represent policies that apply to Managed Elements and optionally define specific
181 CQL or CIM-SPL statements that represent the policies. The *Policy Profile* is to be specialized by profiles
182 that define specific policy algorithms and optionally define CQL or CIM-SPL statements for particular
183 policies.

184 ## 2 Normative References

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

188 ### 2.1 Approved References

189 DMTF DSP0004, *CIM Infrastructure Specification 2.3.0*

190 DMTF DSP0200, *CIM Operations over HTTP 1.2.0*

191 DMTF DSP1000, *Management Profile Specification Template*

192 DMTF DSP1001, *Management Profile Specification Usage Guide*

193

194 ### 2.2 References under Development

195 DMTF DSP1033, *Profile Registration Profile*

196 DMTF DSP0202, *CIM Query Language Specification*

197 DMTF DSP0231, *CIM Simplified Policy Language*

198

199 ### 2.3 Other References

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

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

203 ## 3 Terms and Definitions

204 For the purposes of this document, the following terms and definitions apply. For the purposes of this
205 document, the terms and definitions given in DSP1033 and DSP1001 also apply.

206 **3.1**
207 **can**
208 used for statements of possibility and capability, whether material, physical, or causal

3.2
cannot
used for statements of possibility and capability, whether material, physical, or causal

3.3
conditional
indicates requirements to be followed strictly to conform to the document when the specified conditions are met

3.4
mandatory
indicates requirements to be followed strictly to conform to the document and from which no deviation is permitted

3.5
may
indicates a course of action permissible within the limits of the document

3.6
need not
indicates a course of action permissible within the limits of the document

3.7
optional
indicates a course of action permissible within the limits of the document

3.8
referencing profile
indicates a profile that owns the definition of this class and can include a reference to this profile in its "Referenced Profiles" table

3.9
shall
indicates requirements to be followed strictly to conform to the document and from which no deviation is permitted

3.10
shall not
indicates requirements to be followed strictly to conform to the document and from which no deviation is permitted

3.11
should
indicates that among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required

3.12
should not
indicates that a certain possibility or course of action is deprecated but not prohibited

3.13
unspecified
indicates that this profile does not define any constraints for the referenced CIM element or operation

3.14
policy language
refers to CQL and CIM-SPL as the two defined languages

4 Symbols and Abbreviated Terms

4.1
CQL
CIM Query Language

4.2
CIM-SPL
CIM Simplified Policy Language

5 Synopsis

Profile Name: *Policy*

Version: 1.0.0e

Organization: DMTF

CIM schema version: 2.12

Central Class: CIM_PolicyRule

Scoping Class: CIM_System

This abstract profile specification shall not be directly implemented; implementations shall be based on a profile specification that specializes the requirements of this profile.

The *Policy Profile* is an abstract profile that extends the management capability of referencing profiles by adding the capability to represent policies that apply to Managed Elements and optionally define specific CQL or CIM-SPL statements that represent the policies. The profile is to be specialized by profiles that define specific policy algorithms and optionally define CQL or CIM-SPL statements for particular policies.

Table 1 lists profiles upon which this profile has a dependency.

Table 1 – Referenced Profiles

Profile Name	Organization	Version	Relationship	Behavior

6 Description

The *Policy Profile* is an abstract profile that defines the CIM elements and constraints for specifying policies that affect Managed Elements. Architecture/implementation details are out of the scope of this abstract profile.

A *policy* defines actions that are taken under a set of conditions. Generally, a policy rule is automated. A policy defines what is to be done in the users stead and under what circumstances. The policy rule itself is generally not the entity that carries out the action. Instead, a policy rule is assumed to be the entity that directs the change. Just as when an end user requests that a host reboot, that user is not the entity that

informs the host hardware to reboot directly; policy rules request that the action be taken on its behalf. As such, a policy rule reacts to its own environment.

A condition is expressed as the state of one or more Managed Elements. The description of the state could be simple or complex. It can be described in words or could be defined in terms of values of the properties of the CIM instances.. Regardless of the complexity of the state in the condition, the condition itself defines state criteria that are considered as a whole. If the states of the Managed Elements are as defined in the condition, then the condition itself is evaluated as TRUE. Conditions could be notified of a state change that is triggered.

An *action* defines how the state ought to change as a result of the action itself being invoked. A policy action defines that a single action is taken. Actions can succeed or fail. The action could be described in words or defined in terms of the properties or methods of CIM instances.

A policy rule is evaluated. A policy rule can be assumed to be executed periodically. A PolicyRule may be associated with one or more policy time periods, indicating the schedule according to which the policy rule is active. At all times that fall outside these time periods, the PolicyRule is inactive. A PolicyRule is treated as active at ALL times, if it does not specify a PolicyTimePeriodCondition. The policy rule is executed by a special type of policy that depends on time or could be executed on a schedule defined elsewhere. Otherwise, a policy rule could be evaluated as a result of its trigger conditions being notified of a state change.

During the policy evaluation, the states defined in the associated conditions are compared to the actual states of the related Managed Elements. Each condition is considered in order. The order could be defined explicitly or implicitly. Implicitly ordered conditions are those that depend on the output of the evaluation of other conditions in order to be evaluated. If enough conditions are evaluated to be TRUE, the actions are then taken. Likewise, actions could be ordered explicitly or implicitly. Implicitly ordered actions are those that require the evaluation of conditions or execution of actions for themselves to be executed.

Policy rules must be defined in terms of a response. The policy rule must be defined without depending on other policy rules.

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

Functionality within the scope of this profile includes:

- a specification of the CIM_PolicyRule class and optional related CIM_PolicyCondition and CIM_PolicyAction classes

Functionality explicitly excluded from the scope of this profile includes how the specific CQL or CIM-SPL definitions for policy conditions and actions might be modified by the client. This is defined further in specialized profile.

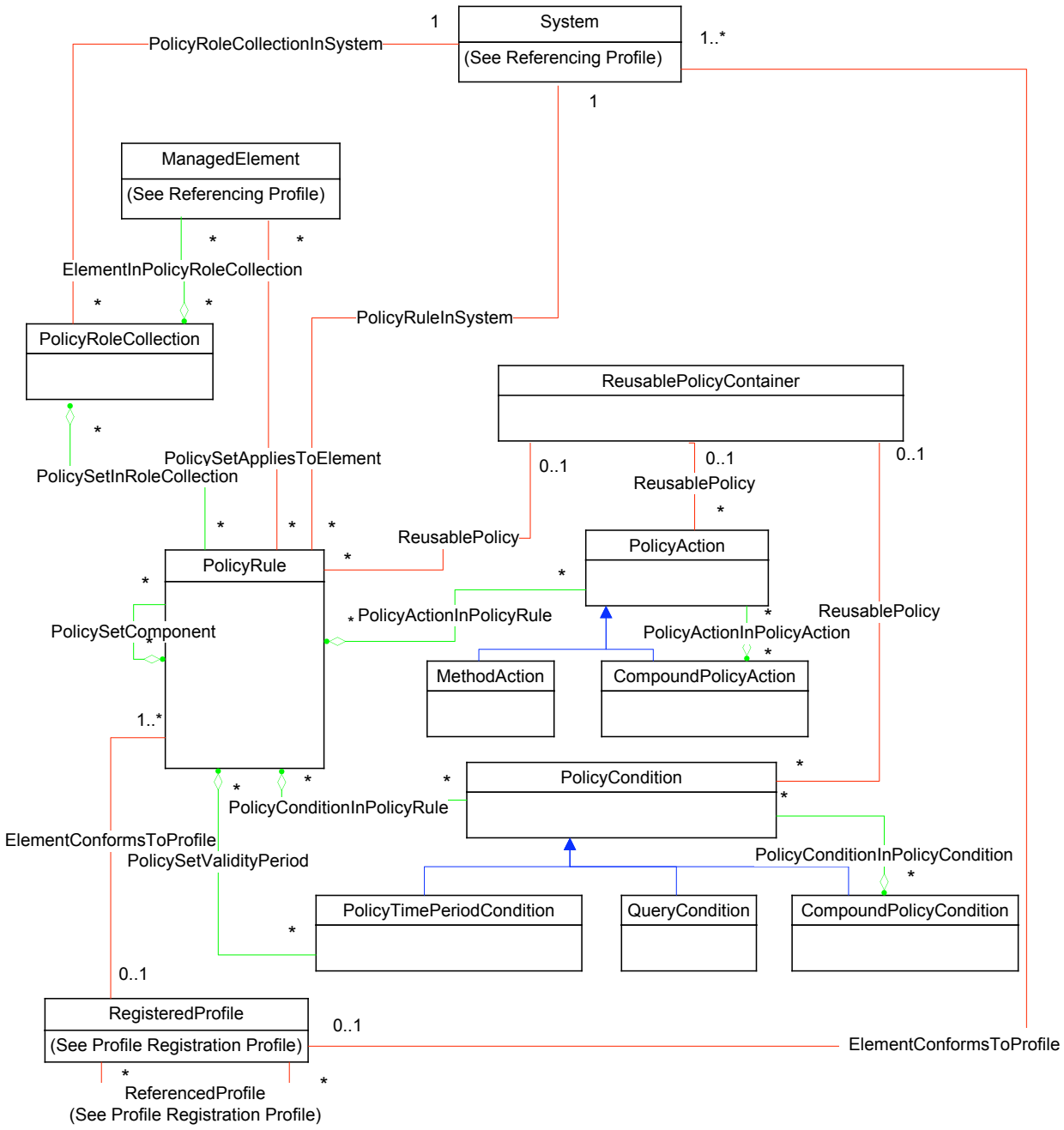


Figure 1 – Policy Profile: Class Diagram

6.1 Whitebox Policies

The term “Whitebox” is applied to policies that have policy rules that can be inspected. Whitebox policy rules are defined in terms of policy language statements that directly interact with the model. As such, special subclasses of CIM_PolicyCondition and CIM_PolicyAction, CIM_QueryCondition and CIM_MethodAction respectively, are used to contain the expression of how the CIM representation or state of the model causes the policy to act, and what CIM action is imposed on the model as a result.

The benefit of Whitebox policies is that they can be inspected, they are programmatically deterministic, and they have precisely the same meaning and result to observers without any prior knowledge. Given

the CIM and the CQL standard and this definition of the behavior and rules of policies, the policy rule algorithms as expressed in CQL can be interpreted by both the CIM clients that read them and the systems that host and evaluate them.

A CIM_QueryCondition instance contains a CQL statement that itself expresses the state whose attainment results in the condition being evaluated as TRUE. When such a CQL statement is evaluated, the CQL is executed and a result is produced. This result is available to other conditions and actions that are members of the policy rule. If the Query Condition produces an empty result, it is considered to be evaluated as FALSE. The result itself can be referenced as a CIM class in CQL construct by using its QueryResultName property.

A CIM_QueryCondition may be dependent on the results of other queries. Specifically, a condition may be dependent on other conditions. This result may be joined with other classes or other results to produce additional results. A dependence of several conditions might be produced such that the set resembles a decision tree. The least dependent conditions must be evaluated first. The most dependent conditions must be evaluated last.

A CIM_MethodAction instance defines the query that itself builds an intrinsic or extrinsic method call. The components of this query must contain first the object name of the method to be called and, thereafter, a list of named parameters. The parameter names and types must match those of the method declaration. Type conversion is supplied by the associated CQL implementation. The result of the selected list applies to the referenced method as input parameters. If the result of the query produces an empty result, the method will not be executed. Such an execution must be considered successful. The method is executed once for every row of the results, which in turn provides the parameters for each method call.

When the CIM_MethodAction query executes, it must apply its action to the model. The result of the method execution shall be the list of output parameters for the method; each, in turn, shall be identifiable by its parameter name.

The query result of the conditions and the method result of the action may be used by other actions to construct method calls.

Such policies can take several execution paths based on the conditions that are evaluated as TRUE and the actions taken as a result. The effect of some of the conditions evaluated as TRUE may result in only some of the actions being taken. For a given policy rule, a different set of conditions may produce a different set of actions depending on the evaluation.

6.2 Blackbox Policies

The “Blackbox” term is applied to policies that have condition and action algorithms defined but cannot be inspected through the CIM model. Blackbox policy rules are represented in CIM by policy rule names or labels, but they are not accompanied by classes that contain specific condition or action query strings. The model manipulation algorithms that represent a Blackbox policy is defined in a profile that describes the algorithms in sufficient detail for consistent implementation by instrumentation.

Determining whether a policy rule represents a blackbox policy can be accomplished by inspecting whether an instance of CIM_RegisteredProfile, which specifies a blackbox policy is associated with an instance of CIM_PolicyRule, which represents the policy rule in question, via the CIM_ElementConformsToProfile association.

7 Implementation

This section details requirements and guidelines for propagating and formulating certain properties of the classes of this profile. Required methods are described in section 8 (“Methods”), and properties are described in section 10 (“CIM Elements”).

7.1 CIM_QueryCondition (Optional)

Instances of CIM_ QueryCondition are optional and may be used to define the criteria for generating a set of query results that are accessible to other QueryConditions or MethodActions of the same PolicyRule.

7.2 CIM_MethodAction (Optional)

Instances of CIM_ MethodAction are optional and may be used to invoke methods as defined by a query.

7.3 CIM_CompoundPolicyAction (Optional)

Instances of CIM_CompoundPolicyAction are optional and may be used to represent an expression that consists of an ordered sequence of actions represented by instances of the CIM_PolicyAction class.

The ordering of action terms shall be represented by instances of the CIM_PolicyActionInPolicyAction aggregation.

7.3.1 Action Sequences

Each action in an explicit action sequence is represented by a CIM_PolicyAction instance aggregated by a CIM_PolicyActionInPolicyAction instance.

The order of the actions in the sequence shall be specified by the ActionOrder property of each CIM_PolicyActionInPolicyAction aggregation instance. An ActionOrder property value of 0 shall mean that the aggregated CIM_PolicyAction instance may be executed at any time in the execution sequence. Non-zero values of ActionOrder indicate a relative priority, in which lower numbers shall be executed before higher numbers. Two or more instances of CIM_PolicyAction aggregated with the same non-zero value may be executed in any order, but after all instances associated with lower non-zero values and before all instances with higher non-zero values.

The optional SequencedActions property of CIM_CompoundPolicyAction specifies the order of execution of the actions in the sequence.

The optional ExecutionStrategy property of CIM_CompoundPolicyAction defines the strategy to be used in executing the action sequence.

7.4 CIM_CompoundPolicyCondition (Optional)

Instances of CIM_CompoundPolicyCondition are optional and shall be used to represent a condition expression. Instances of CIM_PolicyCondition are composed into a condition expression by instances of the CIM_PolicyConditionInPolicyCondition aggregation.

The CIM_PolicyCondition instances shall be aggregated into an instance of CIM_CompoundPolicyCondition forming a set of conditions.

ConditionListType is an optional property of a CIM_CompoundPolicyCondition instance. It is used with the GroupNumber and ConditionNegated properties of each CIM_PolicyConditionInPolicyCondition aggregation instance to collectively specify an explicit condition expression. The set of individual terms of an explicit condition expression is represented by the aggregated CIM_PolicyCondition instances.

The set of terms is divided into one or more subsets. Two instances of CIM_PolicyCondition that belong to the same explicit condition expression shall have the same value for the CIM_PolicyConditionInPolicyCondition.GroupNumber property for the instances of

CIM_PolicyConditionInPolicyCondition that associate them to an instance of CIM_CompoundPolicyCondition.

The ConditionNegated property specifies whether the associated condition must be negated when evaluated in the condition expression.

The ConditionListType property shall have one of the following two values:

- Disjunctive Normal Form (DNF): Each term is evaluated for TRUE or FALSE. If the ConditionNegated property is TRUE for the term, the term's evaluation result is negated. Then, the evaluation result for each term that belongs to a particular subset (the same GroupNumber) is joined together with AND. Finally the result of evaluating each subgroup is joined together with OR.
- Conjunctive Normal Form (CNF): Each term is evaluated for TRUE or FALSE. If the ConditionNegated property is TRUE for the term, the term's evaluation result is negated. Then, the evaluation result for each term that belongs to a particular subset (the same GroupNumber) is joined together with OR. Finally the result of evaluating each subgroup is joined together with AND.

The Disjunctive Normal Form (DNF) is the default value for ConditionListType.

7.5 CIM_ElementInPolicyRoleCollection (Optional)

An instance of CIM_ElementInPolicyRoleCollection is an optional aggregation that may be used to associate a CIM_ManagedElement instance with an instance of CIM_PolicyRoleCollection.

Use of this association shall indicate that instances of CIM_PolicySet, which are aggregated through CIM_PolicySetInRoleCollection into the referenced instance of CIM_PolicyRoleCollection, may be applied to the referenced instance of CIM_ManagedElement.

Constraints on the behavior of the policy as applied to instances of CIM_ManagedElement may be further specified in a specialized policy profile.

7.6 CIM_PolicyAction (Optional)

Instances of the abstract class CIM_PolicyAction are optional. Each instance represents an explicit specification of an action.

CIM_PolicyAction instances may be aggregated by CIM_PolicyActionInPolicyRule into one or more instances of CIM_PolicyRule or by CIM_PolicyActionInPolicyAction into one or more instances of CIM_CompoundPolicyAction. These associations, together with the aggregating instances, define the execution order of an aggregated set of CIM_PolicyAction instances.

If a CIM_PolicyAction instance may be reused, it shall be aggregated into one CIM_ReusablePolicyContainer instance.

7.7 CIM_PolicyActionInPolicyAction (Optional)

Depending on rules within the aggregating CIM_CompoundPolicyAction instance, the execution sequence of the set of aggregated CIM_PolicyAction instances may be enforced. In that case, the value of the optional ActionOrder property of each instance affects the execution sequence of the aggregated CIM_PolicyAction instance. If the ActionOrder property is not specified, or if it is specified with a value of NULL, the property shall have the value 0.

If the order of action is enforced, the execution sequence covers the set of CIM_PolicyAction instances that are associated through CIM_PolicyActionInPolicyAction to the same instance of CIM_CompoundPolicyAction. In this case, a value of 0 shall mean that the aggregated CIM_PolicyAction

instances may be executed at any time in the execution sequence. Non-zero values of ActionOrder indicate a relative priority, in which lower numbers shall be executed before higher numbers. Two or more instances of CIM_PolicyAction aggregated with the same non-zero value may be executed in any order, but after all instances associated with lower non-zero values and before all instances with higher non-zero values.

7.8 CIM_PolicyActionInPolicyRule (Optional)

An instance of CIM_PolicyActionInPolicyRule is an optional aggregation that may be used to explicitly specify that an action, represented by an instance of CIM_PolicyAction, shall be executed if the conditions specified for the referenced instance of CIM_PolicyRule are met.

Depending on rules within the aggregating CIM_PolicyRule instance, the execution sequence of the set of aggregated CIM_PolicyAction instances may be enforced. In that case, the value of the optional ActionOrder property of each instance affects the execution sequence of the aggregated CIM_PolicyAction instances. If the ActionOrder property is not specified, or if it is specified with a value of NULL, the property shall have the value 0.

If the order of action is enforced, the execution sequence covers the set of CIM_PolicyAction instances that are associated by CIM_PolicyActionInPolicyRule to the same instance of CIM_PolicyRule. In this case, a value of 0 shall mean that the aggregated CIM_PolicyAction instances may be executed at any time in the execution sequence. Non-zero values of ActionOrder indicate a relative priority, in which lower numbers shall be executed before higher numbers. Two or more instances of CIM_PolicyAction aggregated with the same non-zero value may be executed in any order, but after all instances associated with lower non-zero values and before all instances with higher non-zero values.

7.9 CIM_PolicyCondition (Optional)

Instances of the abstract class CIM_PolicyCondition are optional. Each instance represents an explicit specification of a condition.

CIM_PolicyCondition instances may be aggregated by CIM_PolicyConditionInPolicyRule into one or more instances of CIM_PolicyRule or by CIM_PolicyConditionInPolicyCondition into one or more instances of CIM_CompoundPolicyCondition. These associations, together with the aggregating instances, define an expression made up of a set of condition terms represented by CIM_PolicyCondition instances.

If a CIM_PolicyCondition instance may be reused, it shall be aggregated into one CIM_ReusablePolicyContainer instance.

7.10 CIM_PolicyConditionInPolicyCondition (Optional)

The GroupNumber is an optional property used to produce subsets of the set of condition terms. If the GroupNumber property is not specified, or if it is specified with a value of NULL, the property shall have the value 0.

ConditionNegated is an optional property used to indicate whether the aggregated CIM_PolicyCondition instance should be negated (TRUE) or not (FALSE). If the ConditionNegated property is not specified, or if it is specified with a value of NULL, the property shall have the value FALSE.

7.11 CIM_PolicyConditionInPolicyRule (Optional)

The CIM_PolicyCondition instances aggregated into an instance of CIM_PolicyRule form a set of condition terms.

GroupNumber is an optional property that may be used to produce subsets of the set of condition terms. If the GroupNumber property is not specified, or if it is specified with a value of NULL, the property shall have the value 0.

ConditionNegated is an optional property used to indicate whether the aggregated CIM_PolicyCondition instance shall be negated (TRUE) or not (FALSE). If the ConditionNegated property is not specified, or if it is specified with a value of NULL, the property shall have the value FALSE.

7.12 CIM_PolicyRoleCollection (Optional)

Instances of CIM_PolicyRoleCollection are optional collections of CIM_PolicyRule instances that collectively meet a particular set of policy roles within a system.

These policy roles shall be represented using the Role Property. The value shall have the form: <RoleName>["&&"<RoleName>]*, where the individual role names appear in alphabetical order (according to the collating sequence for UCS-2). Implementations may treat Role property values that are specified as “role combinations” as simple strings.

Each instance of CIM_PolicyRoleCollection shall be associated with exactly one CIM_System instance through an instance of CIM_PolicyRoleCollectionInSystem.

Each CIM_PolicyRule instance required to meet a particular policy role shall be aggregated into an instance of CIM_PolicyRoleCollection by an instance of CIM_PolicySetInRoleCollection.

7.13 CIM_PolicyRoleCollectionInSystem (Conditional)

An instance of CIM_PolicyRoleCollectionInSystem is a conditional association that shall be used to establish a relationship between an instance of CIM_PolicyRoleCollection and a Scoping Instance, such as an AdminDomain or ComputerSystem. When the optional behavior specified in section 7.13 is supported, the instance of this class is used to associate the in an instance of CIM_PolicyRoleCollection to a Scoping instance.

7.14 CIM_PolicyRule

CIM_PolicyRule is the central class of this profile and at least one instance is mandatory.

CIM_PolicyRule instances represent 'If <condition expression> then {<action sequence>}' semantics.

The condition expression and action sequence of a CIM_PolicyRule may be specified implicitly, explicitly, or both.

Each instance of CIM_PolicySetValidityPeriod creates an implicit condition on the referenced CIM_PolicyRule. Evaluation of that rule is conditional on the referenced CIM_PolicyTimePeriodCondition.

7.14.1 Condition Expressions

The set of individual terms of an explicit condition expression are represented by CIM_PolicyCondition instances that are aggregated by CIM_PolicyConditionInPolicyRule instances.

The ConditionListType property of a CIM_PolicyRule instance together with the GroupNumber and ConditionNegated properties of each CIM_PolicyConditionInPolicyRule aggregation instance collectively specify an explicit condition expression.

The set of terms is divided into one or more subsets. Each subset is defined by a different value for the GroupNumber property.

7.14.2 Evaluation Schedule

A CIM_PolicyRule may also be associated with a set of CIM_PolicyTimePeriodCondition instances that are each aggregated by an instance of CIM_PolicySetValidityPeriod. Collectively, this set specifies the schedule according to which the CIM_PolicyRule is active and inactive.

7.14.3 Rule Composition

An instance of CIM_PolicyRule may use a rule set defined by instances of the CIM_PolicySetComponent association to associate CIM_PolicyRule instances. Such a rule is known as a composite rule.

The implicit and explicit conditions of the aggregating rule apply to each of the subordinate rules.

Any side effects of condition evaluation or the execution of actions shall not affect the result of the evaluation of other conditions evaluated. That is, the conditions in the composite rule may be evaluated in any order before applying the priority and determining which actions are to be executed.

If the conditions of the composite rule are met, its action sequence shall be executed after all applicable actions of subordinate rules are executed.

The Priority property of CIM_PolicySetComponent defines the order of execution of CIM_PolicyRule instances in a rule set. The value of Priority shall be a unique, non-negative integer value, relative to all Priority values in the rule set. The actions of an aggregated CIM_PolicyRule instance with a numerically higher value of Priority shall be executed before the actions of an aggregated instance with a lower Priority value.

PolicyDecisionStrategy is a conditional property that shall be used in a composite rule to define the evaluation method used for policies contained in its rule set. The PolicyDecisionStrategy property is conditional on instance of CIM_PolicyRule being referenced by an instance of CIM_PolicySetComponent.

7.15 CIM_PolicyRuleInSystem

CIM_PolicyRuleInSystem instances are a mandatory scoping association that links a CIM_PolicyRule instance to the CIM_System instance in whose scope the rule is defined.

7.16 CIM_PolicySetAppliesToElement (Optional)

An instance of CIM_PolicySetAppliedToElement may be used to define what managed elements can be affected by the successful execution of the policy. A specialization of this profile may use other mechanism, like CQL or CPL, to define the managed elements that can be affected by the policy. If a specialization of this profile relies on this association, then that profile shall define how the association is used to define the set of managed element affected by policy, accounting for the interaction of that definition with other set definition mechanisms like CPL if used.

7.17 CIM_PolicySetComponent (Optional)

A CIM_PolicySetComponent instance is an optional aggregation used to create a composite rule. The set of aggregated CIM_PolicyRule instances forms a rule set that is evaluated in the order specified by the mandatory Priority property.

The Priority property of CIM_PolicySetComponent is a mandatory property defines the order of execution of CIM_PolicyRule instances in a rule set.

7.18 CIM_PolicySetInRoleCollection (Optional)

An instance of CIM_PolicySetInRoleCollection is an optional aggregation used to define a set of CIM_PolicyRule instances that are necessary to satisfy a policy role defined by the referenced CIM_PolicyRoleCollection instance.

7.19 CIM_PolicySetValidityPeriod (Optional)

An instance of the optional CIM_PolicySetValidityPeriod aggregation associates an instance of CIM_PolicyTimePeriodCondition with an instance of CIM_PolicySet.

A Time Period may be aggregated by multiple CIM_PolicySets. A set that does not point to a CIM_PolicyTimePeriodCondition through this association, from the scheduling perspective, is always in a valid time period.

7.20 CIM_PolicyTimePeriodCondition (Optional)

Each instance of this optional class provides a means of representing the time periods.

One use is to govern when an instance of CIM_PolicyRule is active when aggregated through CIM_PolicySetValidityPeriod. At all times that fall outside these time periods, the aggregating CIM_PolicyRule has no effect. An instance of CIM_PolicyRule is active at all times if it does not aggregate any instances of CIM_PolicyTimePeriodCondition.

A second use of CIM_PolicyTimePeriodCondition is as an explicit term in a condition expression of an instance of CIM_PolicyRule or CIM_CompoundPolicyCondition. Using CIM_PolicyTimePeriodCondition in this way enables the inclusion of time-based criteria in the condition expressions.

The TimePeriod property shall match the pattern "^[0123456789]{8}T[0123456789]{6}/[0123456789]{8}T[0123456789]{6}$".

The value of the optional TimePeriod property is formatted as a string that represents a start date and time, in which the character 'T' indicates the beginning of the time portion, followed by the slash character (/), followed by a similar string that represents an end date and time. The first date indicates the beginning of the range; the second date indicates the end. The second date and time shall be later than the first. Date/times are expressed as substrings of the form yyyymmddThhmmss. For example, 20000101T080000/20000131T120000 defines January 1, 2000, 0800 through January 31, 2000, noon.

In the following special cases, one of the date/time strings may be replaced with a special string defined in RFC 2445:

- If the first date/time is replaced with the string 'THISANDPRIOR', the property indicates that a PolicySet is valid [from now] until the date/time that appears after the slash character (/).
- If the second date/time is replaced with the string 'THISANDFUTURE', the property indicates that a PolicySet becomes valid on the date/time that appears before the slash character (/) and remains valid from that point on.

When the TimePeriod property is NULL, the CIM_PolicyTimePeriodConditon shall evaluate to TRUE.

The value of the optional MonthOfYearMask property refines the valid time period that is defined by the TimePeriod property by explicitly specifying the months during which the condition may evaluate to TRUE. This property is formatted as an octet string, structured as follows:

- A 4-octet length field, indicating the length of the entire octet string. This field is always set to 0x00000006 for this property.
- A 2-octet field consisting of 12 bits identifying the 12 months of the year, beginning with January and ending with December, followed by 4 bits that are always set to 0. For each month, the value 1 indicates that the policy is valid for that month, and the value 0 indicates that it is not valid.

The MonthOfYearMask value 0x000000060830, for example, indicates that the condition evaluates to TRUE only in the months May, November, and December.

If MonthOfYearMask is not specified, or the value is set to NULL, all months of the year are specified for the purpose of evaluating the condition (effectively a value of 0x00000006FFF0).

The value of the optional DayOfMonthMask refines the valid time period that is defined by the TimePeriod property by explicitly specifying the days of the month during which the condition may evaluate to TRUE. This property is formatted as an octet string, structured as follows:

- A 4-octet length field, indicating the length of the entire octet string. This field is always set to 0x0000000C for this property.
- An 8-octet field consisting of 31 bits identifying the days of the month counting from the beginning, followed by 31 more bits identifying the days of the month counting from the end, followed by 2 bits that are always set to 0. For each day, the value 1 indicates that the PolicySet is valid for that day, and the value 0 indicates that it is not valid.

The DayOfMonthMask value 0x0000000C8000000100000000, for example, indicates that a PolicySet is valid on the first and last days of the month.

For months with fewer than 31 days, the digits corresponding to days that the months do not have (counting in both directions) are ignored.

If a DayOfMonthMask value for this property is not provided, or the value is set to NULL, all days of the month are treated as valid for the purpose of evaluating this condition (effectively a value of 0x0000000CFFFFFFFFFFFFFFFC).

The value of the optional DayOfWeekMask refines the valid time period that is defined by the TimePeriod property by explicitly specifying the days of the week during which the condition may evaluate to TRUE. This property is formatted as an octet string, structured as follows:

- A 4-octet length field, indicating the length of the entire octet string.This field is always set to 0x00000005 for this property.
- A 1-octet field consisting of 7 bits identifying the 7 days of the week, beginning with Sunday and ending with Saturday, followed by 1 bit that is always set to 0. For each day of the week, the value 1 indicates that the PolicySet is valid for that day, and the value 0 indicates that it is not valid.

The DayOfWeekMask value 0x000000057C, for example, indicates that a PolicySet is valid Monday through Friday.

If a DayOfWeekMask value for this property is not provided, or the value is set to NULL, all days of the week are treated as valid for the purpose of evaluating this condition (effectively a value of 0x00000005FE).

The TimeOfDayMask property shall match the pattern “^T[0123456789]{6}/T[0123456789]{6} $”.

The value of the optional TimeOfDayMask property refines the valid time period that is defined by the TimePeriod property by explicitly specifying a range of times in a day during which the condition may evaluate to TRUE. This property is formatted in the style of RFC 2445: a time string beginning with the character 'T', followed by the slash character (/), followed by a second time string. The first time indicates the beginning of the range; the second time indicates the end. Times are expressed as substrings of the form 'Thhmmss'.

The second substring always identifies a later time than the first substring. To allow for ranges that span midnight, however, the value of the second substring may be smaller than the value of the first substring. Thus, 'T080000/T210000' identifies the range from 0800 to 2100, while 'T210000/T080000' identifies the range from 2100 to 0800 the following day.

When a range spans midnight, it includes parts of two successive days. When one of these days is also selected by the MonthOfYearMask, DayOfMonthMask, or DayOfWeekMask property, but the other day is not, then the condition evaluates to TRUE only during the portion of the range that falls on the selected day. For example, if the range extends from 2100 to 0800, and the DayOfWeekMask property selects Monday and Tuesday, the PolicySet is active during the following three intervals:

- From midnight Sunday until 0800 Monday
- From 2100 Monday until 0800 Tuesday
- From 2100 Tuesday until 23:59:59 Tuesday

If a TimeOfDayMask value is not provided or the value is set to NULL, the condition may evaluate to TRUE for all hours of the day (effectively a value of 'T000000/T240000').

The value of the optional LocalOrUtcTime property indicates whether the times represented in the TimePeriod property and in the various Mask properties represent local times or UTC times. There is no provision for mixing local times and UTC times; the value of this property applies to all of the other time-related properties. Time periods are synchronized worldwide by using the enumeration value 'UTCTime'. If the goal is to synchronize worldwide on a particular local time (such as 0300 - 0500 in New York), and if the TimePeriod property spans a Daylight Saving Time transition in New York, it will be necessary to create multiple instances of CIM_PolicyTimePeriodCondition, one based on the offset UTC-0500 for the part of each year when standard time is used in New York, and one based on the offset UTC-0400 for the part of each year when Daylight Saving Time is used there.

7.21 CIM_ReusablePolicy (Optional)

Instances of the optional CIM_ReusablePolicy association indicate that the referenced instance of CIM_PolicyRule, CIM_PolicyAction, or CIM_PolicyCondition is a member of an instance of CIM_ReusablePolicyContainer.

7.22 CIM_ReusablePolicyContainer (Optional)

This optional class represents an administratively defined container for reusable instances of CIM_PolicyRule, CIM_PolicyAction, or CIM_PolicyCondition. This class does not introduce any additional properties beyond those in its superclass AdminDomain.

Even if an instance of CIM_PolicyRule and CIM_PolicyCondition is not associated to an instance of CIM_ReusablePolicyContainer through CIM_ReusablePolicy, the instance of CIM_PolicyRule or CIM_PolicyCondition may be aggregated into one or more instances of CIM_PolicyRule.

Each instance of this class shall use the NameFormat value "ReusablePolicyContainer".

8 Methods

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

8.1 CIM_PolicyRoleCollection. ActivatePolicySet()

The mandatory method CIM_PolicyRoleCollection.ActivatePolicySet() takes as input a reference to a CIM_ManagedElement. The CIM_ManagedElement shall be a member of the CIM_PolicyRoleCollection, associated through an instance of ElementInPolicyRoleCollection. The result of this method, if it is successfully executed, is that the aggregated CIM_PolicySets are deployed and enforced for the referenced element. This is reflected by the instantiation of the CIM_PolicySetAppliesToElement association between the referenced element and each instance of CIM_PolicySet.

8.2 CIM_PolicyRoleCollection. DeactivatePolicySet()

The mandatory method CIM_PolicyRoleCollection. DeactivatePolicySet() takes as input a reference to a CIM_ManagedElement. The CIM_ManagedElement shall be a member of the CIM_PolicyRoleCollection, associated through an instance of ElementInPolicyRoleCollection. The result of this method, if it is

708 successfully executed, is that the aggregated CIM_PolicySets are not enforced for the referenced
709 element. This is reflected by the removal of the CIM_PolicySetAppliesToElement association instance, if
710 one exists, between the referenced element and each collected CIM_PolicySet.

711 8.3 Profile Conventions for Operations

712 Support for operations for each profile class (including associations) is specified in the following
713 subclauses. Each subclause includes either the statement "All operations are supported as described by
714 DSP0200 version 1.2" or a table listing all the operations that are not supported by this profile or where
715 the profile requires behavior other than that described by DSP0200 version 1.2.

716 The default list of operations is as follows:

717 • GetInstance
718 • EnumerateInstances
719 • EnumerateInstanceNames
720 • Associators
721 • AssociatorNames
722 • References
723 • ReferenceNames

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

8.4 CIM_QueryCondition

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

8.5 CIM_MethodAction

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

8.6 CIM_PolicyRule

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

8.7 CIM_PolicyCondition

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

8.8 CIM_PolicyAction

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

8.9 CIM_PolicySetAppliesToElement

Table 2 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 2 – Operations: CIM_PolicySetAppliesToElement

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

8.10 CIM_PolicyConditionInPolicyRule

Table 3 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 3 – Operations: CIM_PolicyConditionInPolicyRule

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

8.11 CIM_PolicyActionInPolicyRule

Table 4 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 4 – Operations: CIM_PolicyActionInPolicyRule

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

8.12 CIM_CompoundPolicyAction

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

8.13 CIM_CompoundPolicyCondition

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

8.14 CIM_ElementInPolicyRoleCollection

Table 5 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 5 – Operations: CIM_ElementInPolicyRoleCollection

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

8.15 CIM_PolicyActionInPolicyAction

Table 6llists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 6 – Operations: CIM_PolicyActionInPolicyAction

Operation	Requirement	Messages
EnumerateInstances	Unspecified	None
EnumerateInstanceNames	Unspecified	None
Associators	Unspecified	None

AssociatorNames	Unspecified	None
References	Unspecified	None
ReferenceNames	Unspecified	None

8.16 CIM_PolicyConditionInPolicyCondition

Table 7 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 7 – Operations: CIM_PolicyConditionInPolicyCondition

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

8.17 CIM_PolicyRoleCollection

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

8.18 CIM_PolicyRoleCollectionInSystem

Table 8 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 8 – Operations: CIM_PolicyRoleCollectionInSystem

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

772

773 8.19 CIM_PolicyRuleInSystem

774 Table 9 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or
775 shall not be supported.

776 **Table 9 – Operations: CIM_PolicyRuleInSystem**

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

777 8.20 CIM_PolicySetComponent

778 Table 10 lists operations that either have special requirements beyond those from DSP0200 version 1.2
779 or shall not be supported.

780 **Table 10 – Operations: CIM_PolicySetComponent**

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

781 8.21 CIM_PolicySetInRoleCollection

782 Table 11 lists operations that either have special requirements beyond those from DSP0200 version 1.2
783 or shall not be supported.

784 **Table 11 – Operations: CIM_PolicySetInRoleCollection**

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

8.22 CIM_PolicySetValidityPeriod

Table 12lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 12 – Operations: CIM_PolicySetValidityPeriod

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

8.23 CIM_PolicyTimePeriodCondition

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

8.24 CIM_ReusablePolicy

Table 13 lists operations that either have special requirements beyond those from DSP0200 version 1.2 or shall not be supported.

Table 13 – Operations: CIM_ReusablePolicy

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

8.25 CIM_ReusablePolicyContainer

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

9 Use Cases

This section contains object diagrams and use cases for the *Policy Profile*.

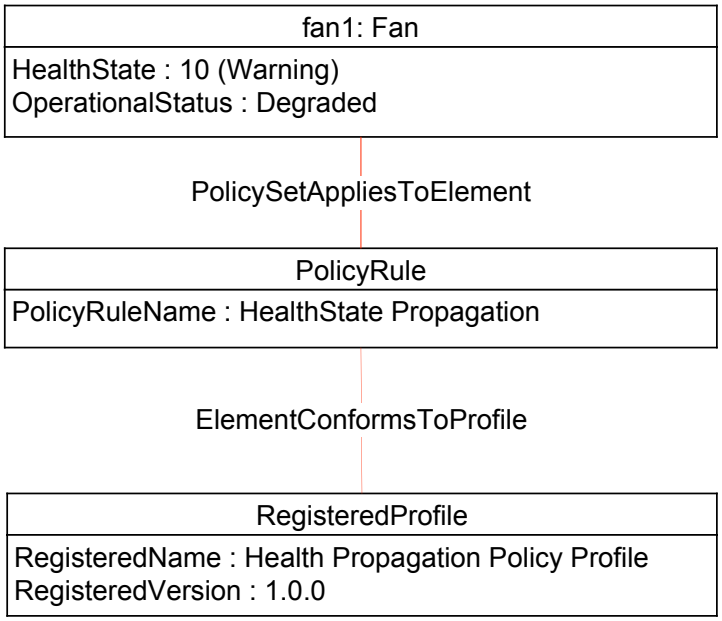
9.1 Object Diagrams

Figure 2 and Figure 3 represent possible policy definitions in a specialized policy profile. The two approaches represented are the Blackbox approach in which the specialized profile defines a policy

804 algorithm, policy name, and policy registered profile. Figure 3 represents taking the definition a step
805 further and defines the actual CQL or CIM-SPL statements that represent the policy algorithm.

806 9.2 Blackbox Policy Object Diagram

807 Figure 2 is an example of a Blackbox definition of CIM_PolicyRule in which the rule represents a policy
808 algorithm that has been defined in a specialized policy profile but does not have specific CQL or CIM-SPL
809 defined.



810

811 **Figure 2 – Blackbox Health Propagation Policy Diagram**

9.3 Propagate HealthState to OperationalStatus

Figure 3 is an example of the CIM_PolicyRule, CIM_PolicyCondition, CIM_PolicyAction, and CIM_RegisteredProfile classes instantiated according to a specialized whitebox policy profile that defines policy for Health State to OperationalStatus propagation within a class.

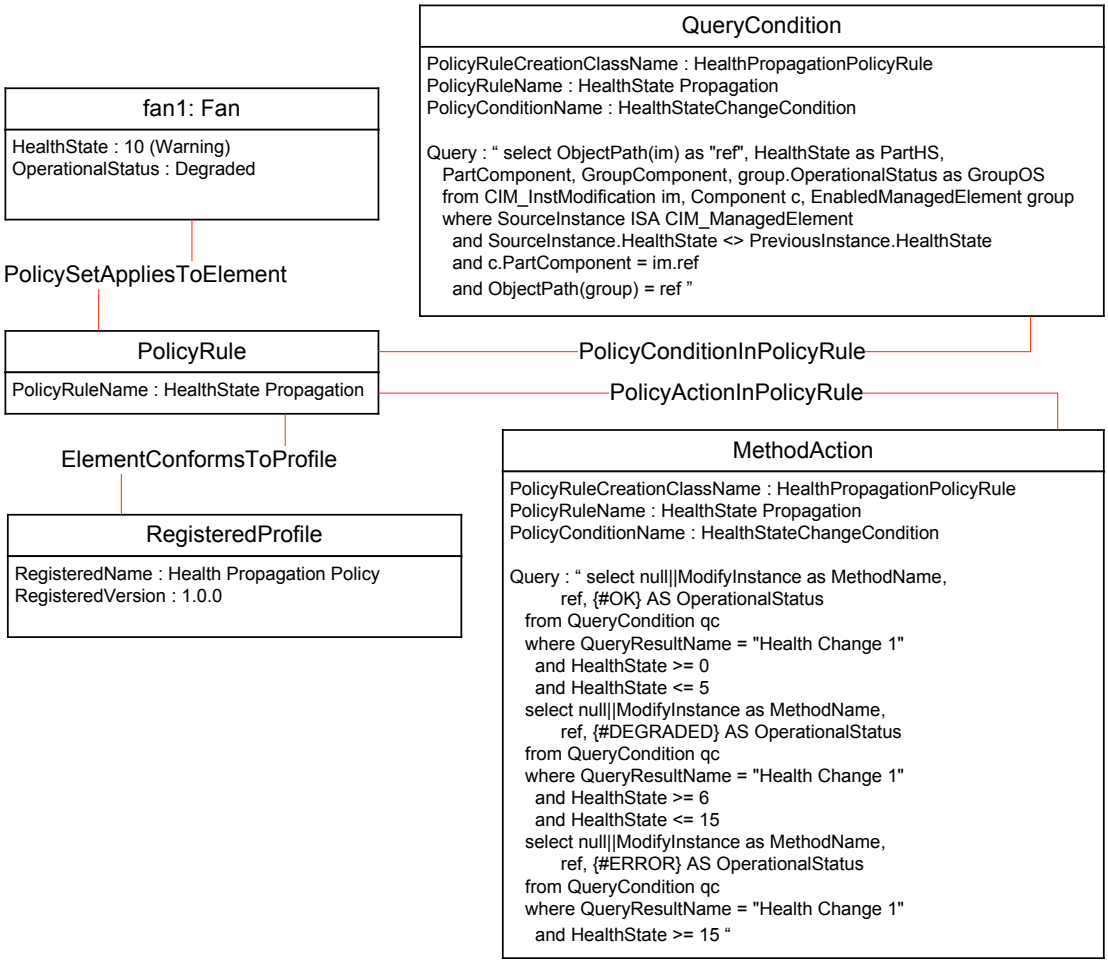


Figure 3 – Health Propagation Policy Object Diagram

10 CIM Elements

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

Table 14 – CIM Elements: Policy Profile

Element Name	Requirement	Description
Classes		
CIM_QueryCondition	Optional	See section 7.1 and 10.1
CIM_MethodAction	Optional	See section 7.2 and 10.2

Element Name	Requirement	Description
CIM_PolicyRule	Mandatory	See section 7.14 and 10.3.
CIM_PolicyCondition	Optional	See section 7.9 and 10.4.
CIM_PolicyAction	Optional	See section 7.6 and 10.5.
CIM_PolicySetAppliesToElement	Optional	See section 7.16 and 10.6
CIM_PolicyConditionInPolicyRule	Optional	See section 7.11 and 10.7
CIM_PolicyActionInPolicyRule	Optional	See section 7.8 and 10.8
CIM_CompoundPolicyAction	Optional	See section 7.1 and 10.9
CIM_CompoundPolicyCondition	Optional	See section 7.4 and 10.10
CIM_ElementInPolicyRoleCollection	Optional	See section 7.5 and 10.11
CIM_PolicyActionInPolicyAction	Optional	See section 7.7 and 10.12
CIM_PolicyConditionInPolicyCondition	Optional	See section 7.10 and 10.13
CIM_PolicyRoleCollection	Optional	See section 7.12 and 10.14
CIM_PolicyRoleCollectionInSystem	Conditional	See section 7.13 and 10.15
CIM_PolicyRuleInSystem	Mandatory	See section 7.15 and 10.16
CIM_PolicySetComponent	Optional	See section 7.17 and 10.17
CIM_PolicySetInRoleCollection	Optional	See section 7.18 and 10.18
CIM_PolicySetValidityPeriod	Optional	See section 7.19 and 10.19
CIM_PolicyTimePeriodCondition	Optional	See section 7.20and 10.20
CIM_ReusablePolicy	Optional	See section 7.21 and 10.21
CIM_ReusablePolicyContainer	Optional	See section 7.22and 10.22
Indications		
No Indications defined in this version		

10.1 CIM_QueryCondition

CIM_QueryCondition defines the criteria for generating a set of query results that are accessible to other QueryConditions or MethodActions of the same PolicyRule. Table 15 contains the requirements for elements of this class.

Table 15 – Class: CIM_QueryCondition

Properties	Requirement	Notes
SystemCreatonClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyConditionName	Mandatory	Key
QueryResultName	Mandatory	This property shall be treated as a class name in a query statement.
Query	Mandatory	This property shall be a query expression that may be evaluated and that defines the query results that may be generated.
QueryLanguage	Mandatory	This property shall be the language defined by DSP0200. The language shall be the one in which the query string is expressed.
Trigger		This property shall be a Boolean condition and there shall be no more than one QueryCondition with this property = true associated with a particular Policy.

10.2 CIM_MethodAction

CIM_ MethodAction is a CIM_PolicyAction that MAY invoke methods as defined by a query. Table 16 contains the requirements for elements of this class.

Table 16 – Class: CIM_MethodAction

Properties	Requirement	Notes
SystemCreatonClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyActionName	Mandatory	Key
InstMethodCallName	Mandatory	This property shall be treated as a class name in a query statement.
Query	Mandatory	This property shall be a query expression that defines the method to invoke and its input parameters.

Properties	Requirement	Notes
QueryLanguage	Mandatory	This property shall be the language defined by DSP0200. The language shall be the one in which the query string is expressed.

832

833 10.3 CIM_PolicyRule

834 CIM_PolicyRule represents the policy rule applied to the associated CIM_ManagedElement class. Table
835 17 contains the requirements for elements of this class.

836 **Table 17 – Class: CIM_PolicyRule**

Properties	Requirement	Notes
SystemCreatonClassName	Mandatory	Key
SystemName	Mandatory	Key
CreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
PolicyDecisionStrategy	Conditional	See section 7.14.3
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

837 10.4 CIM_PolicyCondition

838 CIM_PolicyCondition represents a specific condition defined in CQL or CIM-SPL for triggering a related
839 CIM_PolicyAction. Table 18 contains the requirements for elements of this class.

840 **Table 18 – Class: CIM_PolicyCondition**

Properties	Requirement	Notes
SystemCreationClassName	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
PolicyConditionName	Mandatory	Key
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

10.5 CIM_PolicyAction

CIM_PolicyAction represents a specific action defined in CQL or CIM-SPL to be invoked when a specific policy condition is triggered. Table 19 contains the requirements for elements of this class.

Table 19 – Class: CIM_PolicyAction

Properties	Requirement	Notes
SystemCreationClassName	Mandatory	Key
CreationClassName	Mandatory	Key
SystemName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
PolicyActionName	Mandatory	Key
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

10.6 CIM_PolicySetAppliesToElement

CIM_PolicySetAppliesToElement associates the CIM_PolicySet instance with the CIM_ManagedElement instance whose state is evaluated by policy conditions and may be affected by the successful execution of policy actions. The managed element may be an aggregation point for several other managed element. In this role, the existence of this association to this aggregation point represents the set of instances affected and evaluated by the policy.

Table 20 contains the requirements for elements of this class.

Table 20 – Class: CIM_PolicySetAppliesToElement

Elements	Requirement	Notes
PolicySet	Mandatory	This property shall be a reference to the instance of CIM_PolicySet. Cardinality *
ManagedElement	Mandatory	This property shall be a reference to the instance of CIM_ManagedElement. Cardinality *

10.7 CIM_PolicyConditionInPolicyRule

CIM_PolicyConditionInPolicyRule is an optional aggregation that may be used to compound CIM_PolicyCondition instances into an instance of CIM_PolicyRule.

Table 21 contains the requirements for elements of this class.

861 **Table 21 – Class: CIM_PolicyConditionInPolicyRule**

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to the instance of CIM_PolicySetRule.
PartComponent	Mandatory	This property shall be a reference to the instance of CIM_PolicyCondition.
GroupNumber	Optional	This property is an optional property used to produce subsets of the set of condition terms.
ConditionNegated	Oprional	See section 7.11

862 ## 10.8 CIM_PolicyActionInPolicyRule

863 CIM_PolicyActionInPolicyRule is an optional aggregation that may be used to explicitly specify that an
864 action, represented by an instance of CIM_PolicyAction, shall be executed if the conditions specified for
865 the referenced instance of CIM_PolicyRule are met.

866 Table 22 contains the requirements for elements of this class.

867

868 **Table 22 – Class: CIM_PolicyActionInPolicyRule**

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to the instance of CIM_PolicyRule.
ManagedElement	Mandatory	This property shall be a reference to the instance of CIM_PolicyAction.
ActionOrder	Optional	This property is is an unsigned integer ’ n ’ that indicates the relative position of a PolicyAction in the sequence of actions associated with a PolicyRule or CompoundPolicyAction.

869 ## 10.9 CIM_CompoundPolicyAction

870 CIM_CompoundPolicyAction is an optional class that may be used to represent an expression consisting
871 of an ordered sequence of action terms. Each action term is represented as a subclass of the
872 PolicyAction class. Compound actions are constructed by associating dependent action terms together
873 using the PolicyActionInPolicyAction aggregation.

874 Table 23 contains the requirements for elements of this class.

875

876 **Table 23 – Class: CIM_CompoundPolicyAction**

Elements	Requirement	Notes
SequencedActions	Optional	See Section 7.3.1
ExecutionStrategy	Optional	See Section 7.3
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
CreationClassName	Mandatory	Key
PolicyActionName	Mandatory	Key

877

878 10.10 CIM_CompoundPolicyCondition

879 CIM_CompoundPolicyCondition is an optional class that may be used to represent compound conditions
880 formed by aggregating simpler policy conditions. Compound conditions are constructed by associating
881 subordinate condition terms together using the PolicyConditionInPolicyCondition aggregation.

882 Table 24 contains the requirements for elements of this class.

883

884 **Table 24 – Class: CIM_CompoundPolicyCondition**

Elements	Requirement	Notes
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
CreationClassName	Mandatory	Key
PolicyConditionName	Mandatory	Key
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

885 10.11 CIM_ElementInPolicyRoleCollection

886 CIM_ElementInPolicyRoleCollection is an optional association that may be used to aggregate zero or
887 more Managed Element subclass instances into a PolicyRoleCollection object, representing a role played
888 by these Managed Elements. This Collection indicates that the aggregated PolicySets (aggregated by
889 CIM_PolicySetInRoleCollection) MAY BE applied to the referenced elements.

890 Table 25 contains the requirements for elements of this class.

891

892 **Table 25 – Class: CIM_ElementInPolicyRoleCollection**

Elements	Requirement	Notes
Collection	Mandatory	Key
Member	Mandatory	Key

893

894 10.12 CIM_PolicyActionInPolicyAction

895 CIM_PolicyActionInPolicyAction is an optional association that may be used to represent the
896 compounding of policy actions into a higher-level policy action.

897 Table 26 contains the requirements for elements of this class.

898

899 **Table 26 – Class: CIM_PolicyActionInPolicyAction**

Elements	Requirement	Notes
GroupComponent	Mandatory	Key
PartComponent	Mandatory	Key
ActionOrder	Optional	This property is an unsigned integer ' n ' that indicates the relative position of a PolicyAction in the sequence of actions associated with a PolicyRule or CompoundPolicyAction.

900

901 10.13 CIM_PolicyConditionInPolicyCondition

902 CIM_PolicyConditionInPolicyCondition is an optional association that may be used to aggregate zero or
903 more instances of the PolicyCondition class to a CompoundPolicyCondition.

904 Table 27 contains the requirements for elements of this class.

905

906 **Table 27 – Class: CIM_PolicyConditionInPolicyCondition**

Elements	Requirement	Notes
GroupComponent	Mandatory	Key
PartComponent	Mandatory	Key
GroupNumber	Optional	See Section 7.10
ConditionNegated	Optional	See Section 7.10

907 10.14 CIM_PolicyRoleCollection

908 CIM_PolicyRoleCollection is an optional class that may be used to to represent a collection of Managed
909 Elements that share a common policy role, and the PolicySets that CAN BE applied to those elements.
910 The PolicyRoleCollection always exists in the context of a System, specified using the
911 PolicyRoleCollectionInSystem aggregation. The value of the PolicyRole property in this class specifies

912 the role. It is defined as a free-form string. Managed Elements that share the role defined in this collection
913 are aggregated into the Collection via the ElementInPolicyRoleCollection association.

914 Table 28 contains the requirements for elements of this class.

915

916 **Table 28 – Class: CIM_PolicyRoleCollection**

Elements	Requirement	Notes
PolicyRole	Mandatory	See Section 7.12
InstanceID	Mandatory	Key
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

917 10.15 CIM_PolicyRoleCollectionInSystem

918 CIM_PolicyRoleCollectionInSystem is a conditional association that shall be used to establish a
919 relationship between an instance of CIM_PolicyRoleCollection and an "owning" CIM_System instance,
920 such as an AdminDomain or ComputerSystem.

921 Table 29 contains the requirements for elements of this class.

922

923 **Table 29 – Class: CIM_PolicyRoleCollectionInSystem**

Elements	Requirement	Notes
Antecedent	Mandatory	Key
Dependent	Mandatory	Key

924 10.16 CIM_PolicyRuleInSystem

925 CIM_PolicyRuleInSystem is a mandatory association that shall be used to link a PolicyRule to the System
926 in whose scope the Rule is defined.

927 Table 30 contains the requirements for elements of this class.

928

929 **Table 30 – Class: CIM_PolicyRuleInSystem**

Elements	Requirement	Notes
Antecedent	Mandatory	Key
Dependent	Mandatory	Key

930 10.17 CIM_PolicySetComponent

931 CIM_PolicySetComponent is an optional association that may be used to aggregate the instances of the
932 subclasses of PolicySet (i.e., PolicyGroups and PolicyRules). Instances are collected in sets that use the
933 same decision strategy. They are prioritized relative to each other, within the set, using the Priority
934 property of this aggregation. Together, the PolicySet.PolicyDecisionStrategy and PolicySet
935 Component.Priority properties determine the processing for the groups and rules contained in a
936 PolicySet. A larger priority value represents a higher priority. Note that the Priority property MUST have a

unique value when compared with others defined for the same aggregating PolicySet. Thus, the evaluation of rules within a set is deterministically specified.

Table 31 contains the requirements for elements of this class.

Table 31 – Class: CIM_PolicySetComponent

Elements	Requirement	Notes
GroupComponent	Mandatory	Key
PartComponent	Mandatory	Key
Priority	Mandatory	See section 7.17

10.18 CIM_PolicySetInRoleCollection

CIM_PolicySetInRoleCollection is an optional association that may be used to aggregate zero or more PolicyRules and PolicyGroups (i.e., the subclasses of PolicySet) into a PolicyRoleCollection object, representing a role supported/enforced by the PolicySet.

Table 32 contains the requirements for elements of this class.

Table 32 – Class: CIM_PolicySetInRoleCollection

Elements	Requirement	Notes
Collection	Mandatory	Key
Member	Mandatory	Key

10.19 CIM_PolicySetValidityPeriod

CIM_PolicySetValidityPeriod is an optional association that may be used to aggregate scheduled activation and deactivation of a PolicySet. A PolicySet is considered "active" if it is both "Enabled" and in a valid time period. If a PolicySet is associated with multiple policy time periods via this association, then the Set is in a valid time period if at least one of the time periods evaluates to TRUE. If a PolicySet is contained in another PolicySet via the PolicySetComponent aggregation (e.g., a PolicyRule in a PolicyGroup), then the contained PolicySet (e.g., PolicyRule) is in a valid period if at least one of the aggregate's PolicyTimePeriodCondition instances evaluates to TRUE and at least one of its own PolicyTimePeriodCondition instances also evalutes to TRUE. (In other words, the PolicyTimePeriodConditions are ORed to determine whether the PolicySet is in a valid time period and then ANDed with the ORed PolicyTimePeriodConditions of each of PolicySet instances in the PolicySetComponent hierarchy to determine if the PolicySet is in a valid time period and, if also "Enabled", therefore, active, i.e., the hierachy ANDs the ORed PolicyTimePeriodConditions of the elements of the hierarchy. A Time Period may be aggregated by multiple PolicySets. A Set that does not point to a PolicyTimePeriodCondition via this association, from the point of view of scheduling, is always in a valid time period.

Table 33 contains the requirements for elements of this class.

967 **Table 33 – Class: CIM_PolicySetValidityPeriod**

Elements	Requirement	Notes
GroupComponent	Mandatory	Key
PartComponent	Mandatory	Key

968 ## 10.20 CIM_PolicyTimePeriodCondition

969 CIM_PolicyTimePeriodCondition is an optional class that may be used to represent the time periods
970 during which a PolicySet is valid, i.e., active. At all times that fall outside these time periods, the PolicySet
971 has no effect.

972 Table 34 contains the requirements for elements of this class.

973

974 **Table 34 – Class: CIM_PolicyTimePeriodCondition**

Elements	Requirement	Notes
TimePeriod	Optional	See Section 7.20
MonthOfYearMask	Optional	See Section 7.20
DayOfMonthMask	Optional	See Section 7.20
DayOfWeekMask	Optional	See Section 7.20
TimeOfDayMask	Optional	See Section 7.20
LocalOrUtcTime	Optional	See Section 7.20
SystemCreationClassName	Mandatory	Key
SystemName	Mandatory	Key
PolicyRuleCreationClassName	Mandatory	Key
PolicyRuleName	Mandatory	Key
CreationClassName	Mandatory	Key
PolicyConditionName	Mandatory	Key
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

975 ## 10.21 CIM_ReusablePolicy

976 CIM_ReusablePolicy is an optional association that may be used to establish dependency relationships
977 between Policies and the Systems that host them.

978 Table 35 contains the requirements for elements of this class.

979

980 **Table 35 – Class: CIM_ReusablePolicy**

Elements	Requirement	Notes
Antecedent	Mandatory	Key
Dependent	Mandatory	Key

981

982 10.22 CIM_ReusablePolicyContainer

983 CIM_ReusablePolicyContainer is an optional class that may be used to represente an administratively
984 defined container for reusable policy-related information. This class does not introduce any additional
985 properties beyond those in its superclass AdminDomain. It does, however, participate in a unique
986 association for containing policy elements. An instance of this class uses the NameFormat value
987 "ReusablePolicyContainer".

988 Table 36 contains the requirements for elements of this class.

989

990 **Table 36 – Class: CIM_ReusablePolicyContainer**

Elements	Requirement	Notes
CreationClassName	Mandatory	Key
Name	Mandatory	Key
ElementName	Mandatory	This property shall be formatted as a free-form string of variable length (pattern ".*").

991

992

ANNEX A
(Informative)

Change Log

Version	Date	Description

ANNEX B
(informative)

Acknowledgments

The authors wish to acknowledge the following people.

Editor:

- Jianwen Yin – Dell

Contributors:

- George Ericson – EMC
- Jon Hass – Dell
- Jianwen Yin – Dell
- Steve Hand – Symantec Corp.
- Aaron Merkin – IBM