Dell Lifecycle Controller (LC) Management Profile

Document Number: DCIM1039
Document Type: Specification
Document Status: Published
Document Language: E
Date: 2009-12-16

Version: 1.0.0

THIS PROFILE IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND. ABSENT A SEPARATE AGREEMENT BETWEEN YOU AND DELL™ WITH REGARD TO FEEDBACK TO DELL ON THIS PROFILE SPECIFICATION, YOU AGREE ANY FEEDBACK YOU PROVIDE TO DELL REGARDING THIS PROFILE SPECIFICATION WILL BE OWNED AND CAN BE FREELY USED BY DELL.

© 2008 – 2009 Dell Inc. All rights reserved. Reproduction in any manner whatsoever without the express written permission of Dell, Inc. is strictly forbidden. For more information, contact Dell.

Dell and the *DELL* logo are trademarks of Dell Inc. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Dell disclaims proprietary interest in the marks and names of others.

CONTENTS

1 Scope 5
2 Normative References 5
2.1 Approved References 5
2.2 Other References 5
3 Terms and Definitions 5
4 Symbols and Abbreviated Terms 7
5 Synopsis 7
6 Description 7
7 Implementation Requirements 8
7.1 DCIM_LCService 8
7.2 DCIM_LCEnumeration 8
7.3 DCIM_LCString 8
7.4 DCIM_LCRegisteredProfile 9
7.5 Lifecycle Controller Attributes to be Represented by LC Management Profile 9
8 Methods 10
8.1 DCIM_LCService.SetAttribute() 10
8.2 DCIM_LCService.SetAttributes() 12
8.3 DCIM_LCService.CreateConfigJob() 13
8.4 DCIM_LCService.ReInitiateDHS() 14
8.5 DCIM_LCService.ClearProvisioningServer() 14
8.6 DCIM_LCService.DownloadServerPublicKey() 15
8.7 DCIM_LCService.DownloadClientCerts() 15
9 Use Cases 16
9.1 Object Diagrams 16
9.2 Use Case 1 – Representing CCR, System ID and DHS attributes 17
10 CIM Elements 17
10.1 DCIM_LCAttribute 18
10.2 DCIM_LCService 18
10.3 DCIM_LCEnumeration 18
10.4 DCIM_LCString 20
10.5 DCIM_LCRegisteredProfile 20
ANNEX A (informative) Change Log 20
ANNEX B (informative) Dell Extension MOF 21

Figures

Figure 1 – LC Management Profile: Class Diagram 8
Figure 2 – Object Diagram representing CCR, System ID and DHS attributes 17

Tables

Table 1 – Related Profiles 7
Table 2 – CCR Attributes 9
Table 3 – Discovery and Handshake Attributes 10
Table 4 – DCIM_LCService.SetAttribute() Method: Return Code Values 11
Table 5 – DCIM_LCService.SetAttribute() Method: Parameters 11
Table 6 – DCIM_LCService.SetAttributes() Method: Return Code Values 12
Table 7 – DCIM_LCService.SetAttributes() Method: Parameters 12
Table 8 – DCIM_LCService.CreateConfigJob() Method: Return Code Values 13
Table 9 – DCIM_LCService.CreateConfigJob() Method: Parameters 13
Table 10 – DCIM_LCService.ReInitiateDHS() Method: Return Code Values 14
Table 11 – DCIM_LCService.ReInitiateDHS() Method: Parameters 14
Table 12 – DCIM_LCService.ClearProvisioningServer() Method: Return Code Values 15
Table 13 – DCIM_LCService.ClearProvisioningServer() Method: Parameters 15
Table 14 – DCIM_LCService.DownloadServerPublicKey() Method: Return Code Values 15
Table 15 – DCIM_LCService.DownloadServerPublicKey() Method: Parameters 15
Table 16 – DCIM_LCService.DownloadClientCerts() Method: Return Code Values 16
Table 17 – DCIM_LCService.DownloadClientCerts() Method: Parameters 16
Table 18 – CIM Elements: iDRAC Attribute Profile 17
Table 19 – Class: DCIM_LCAttribute 18
Table 20 – Class: DCIM_LCService 18
Table 21 – Class: DCIM_LCEnumeration 19
Table 22 – Class: DCIM_LCString 20
Table 23 – Class: CIM_RegisteredProfile 20

Lifecycle Controller Management Profile

1 Scope

The Lifecycle Controller (LC) Management Profile extends the management capabilities of referencing profiles by adding the capability to represent the configuration attributes for the Dell Lifecycle Controller. The LC configuration attributes are modeled as attribute collections for an individual LC; typically there is one LC per system platform. This profile is a specialization of the BIOS Management Profile.

2 Normative References

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

2.1 Approved References

DMTF DSP1033, *Profile Registration Profile 1.0.0*

DMTF DSP1061, *BIOS Management Profile 1.0.0*

DELL, *Job Control Profile 1.0.0*

DMTF DSP0200, *CIM Operations over HTTP 1.2.0*

DMTF DSP0004, *CIM Infrastructure Specification 2.3.0*

DMTF DSP1000, *Management Profile Specification Template*

DMTF DSP1001, *Management Profile Specification Usage Guide*

2.2 Other References

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

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

3 Terms and Definitions

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

3.1
can
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 in order to conform to the document when the specified conditions are met

3.4
mandatory
indicates requirements to be followed strictly in order 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 “Related Profiles” table

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

3.10
shall not
indicates requirements to be followed strictly in order 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

4 Symbols and Abbreviated Terms

4.1

CIM

Common Information Model

5 Synopsis

Profile Name: Lifecycle Controller (LC) Management

Version: 1.0.0

Organization: DCIM

CIM Schema Version: 2.19.1

Central Class: DCIM_LCService

Scoping Class: CIM_ComputerSystem

The LC Management Profile extends the management capability of the referencing profiles by adding the capability to describe LC configuration attributes simply. In this profile, LC is represented by a collection and zero or more instances of the CIM_BIOSAttribute class each representing a LC configuration-related attribute. DCIM_LCService shall be the Central Class. CIM_ComputerSystem shall be the Scoping Class. An instance of DCIM_LCService shall be the Central Instance. The instance of CIM_ComputerSystem that the Central Instance is associated with through the CIM_HostedService association shall be the Scoping Instance.

Table 1 identifies profiles that are related to this profile.

Table 1 – Related Profiles

Profile Name	Organization	Version	Relationship
Profile Registration Profile	DMTF	1.0	Mandatory
Dell Job Control	DCIM	1.0	Mandatory

6 Description

The LC Management Profile describes the LC attribute configuration service and the collections and attributes instances that the service manages. The profile also describes the relationship of the LC attribute service to the DMTF/Dell profile version information and Dell Job Control profile.

Figure 1 represents the class schema for the LC Management Profile. For simplicity, the prefix CIM_ has been removed from the names of the classes. The LC service in a managed system is represented by the instance of the DCIM_LCService class. The LC is represented by the CIM_ConcreteCollection class.

The LC Management Profile information is represented with the instance of the CIM_RegisteredProfile.

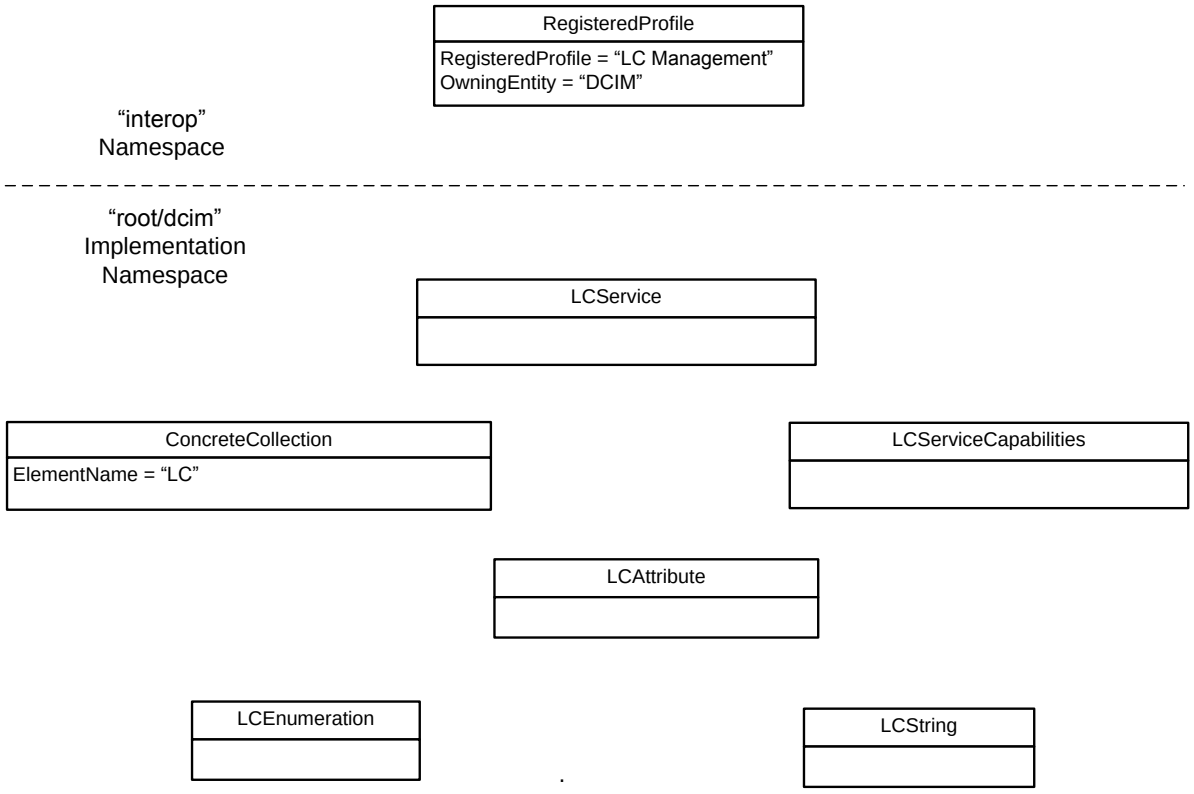

Figure 1 – LC Management Profile: Class Diagram

7 Implementation Requirements

Requirements and guidelines for propagating and formulating certain class properties are discussed in this section; methods are listed in section 8, and properties are listed in section 10.

7.1 DCIM_LCService

One instance of DCIM_LCService shall be instantiated.

7.2 DCIM_LCEnumeration

One or more instances of DCIM_LCEnumeration shall be instantiated.

7.3 DCIM_LCString

One or more instances of DCIM_LCString shall be instantiated.

7.4 DCIM_LCRegisteredProfile

One instance of DCIM_LCRegisteredProfile shall be instantiated.

7.5 Lifecycle Controller Attributes to be Represented by LC Management Profile

The LC Management Profile defines the modeling of LC configuration attributes using the derivation of CIM_BIOSAttribute class to represent each LC configuration attribute. The following are the LC configuration areas that may require representation using this profile. Some areas are already represented using other profiles, and will not need representation using the CIM_BIOSAttribute classes. Each of the attributes represented by the derivation of the CIM_BIOSAttribute class may be modified using SetAttribute() and SetAttributes() methods.

7.5.1 CCR (Component Configuration Recovery) Attributes:

- Licensed
- Part Firmware Update
- Collect System Inventory on Restart

Table 2 – CCR Attributes

Attribute Name	Possible Values
Licensed	"Yes" = Licensed "No" = Not Licensed
Part Firmware Update	Disable (default) = firmware update is not allowed Allow version upgrade only = Allow firmware update only on up-revision Match firmware of replaced part = Always update firmware
Collect System Inventory on Restart	Disabled (default) = Disallow collecting inventory on restart Enabled = Allow collecting system inventory on restart

7.5.2 Dell System ID

- System ID

7.5.3 Discovery and Handshake Attributes:

- Provisioning Server
- Auto Discovery
- Discovery Factory Defaults

Table 3 – Discovery and Handshake Attributes

Attribute Name	Possible Values
Provisioning Server	provisioning server addresses and ports Examples: hostname hostname.domain.com 1.1.1.1 Mypc:8080 Myps.dell.com(1.2.3.4):8080 Host1:80,host2:8080 Hostname[2001:db8:0000:1428:57ab]:443 [2001:db8:0000:1428:57ab]:443
Auto Discovery	Off (default) = Auto Discovery feature is turned off On = Auto Discovery feature is turned on
Discovery Factory Defaults	Off (default) = Do not reset to factory defaults when performing auto discovery On = Reset to factory defaults when performing auto discovery

8 Methods

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

8.1 DCIM_LCService.SetAttribute()

A method used to set the value of a single attribute.

Table 4 – DCIM_LCService.SetAttribute() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred

Table 5 – DCIM_LCService.SetAttribute() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	AttributeName	string	DCIM_LCAttribute.AttributeName
IN	AttributeValue	string [ ]	Pending or Current value to be set
OUT	SetResult	string	Invoking the SetAttribute may result in the CurrentValue or PendingValue property being set. A value of “2” means CurrentValue property is set, and a value of “3” means PendingValue property is set
OUT	RebootRequired	string	A value of “Yes” means a reboot is required to set this value, and a value of “No” means a reboot is not required to set this value
OUT (optional)	MessageID	string	Error Message ID- can be used to index into Dell Message registry files
OUT (optional)	Message	string	Error Message in English corresponding to MessageID is returned if the method fails to execute
OUT	MessageArguments	string[]	Substitution variables for dynamic error messages

8.2 DCIM_LCService.SetAttributes()

A method used to set the values of multiple attributes.

Table 6 – DCIM_LCService.SetAttributes() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred

Table 7 – DCIM_LCService.SetAttributes() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	AttributeName	string [ ]	Array of DCIM_LCAttribute.AttributeName
IN	AttributeValue	string [ ]	Corresponding array of Pending or Current value to be set
OUT	SetResult	string [ ]	invoking the SetAttributes may result in the CurrentValue or PendingValue property of each input element being set. AttributeValue array will have a corresponding SetResult value in the SetResult array. A value of “2” means the CurrentValue property is set, and a value of “3” means the PendingValue property is set
OUT	RebootRequired	string [ ]	Each input element‘s AttributeValue array will have a corresponding RebootRequired value in the RebootRequired array. A value of “Yes” means a reboot is required to set this value, and a value of “No” means a reboot is not required to set this value
OUT	MessageID	string [ ]	Error Message ID- can be used to index into Dell Message registry files
OUT	Message	string [ ]	Error Message in English corresponding to MessageID is returned if the method fails to execute
OUT	MessageArguments	string[]	Substitution variables for dynamic error messages

8.3 DCIM_LCService.CreateConfigJob()

A method used to apply the pending values set by the SetAttribute and SetAttributes methods.

Table 8 – DCIM_LCService.CreateConfigJob() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

Table 9 – DCIM_LCService.CreateConfigJob() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN (optional)	ScheduledStartTime	Datetime	Job will be scheduled immediately. Scheduling is not supported in this version
IN (optional)	RebootIfRequired	Boolean	Reboot the system if it is required for the current value to be effective.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status
OUT	MessageID	string	Error Message ID- can be used to index into Dell Message registry files
OUT	Message	string	Error Message in English corresponding to MessageID is returned if the method fails to execute
OUT	MessageArguments	string[]	Substitution variables for dynamic error messages

8.4 DCIM_LCService.ReInitiateDHS()

A method used to reinitiate the provisioning server discovery and handshake.

Table 10 – DCIM_LCService.ReInitiateDHS() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred

Table 11 – DCIM_LCService.ReInitiateDHS() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN (optional)	ProvisioningServer	string	An optional parameter to specify provisioning server addresses and ports used for auto discovery. If omitted, the Lifecycle Controller will get the value from DHCP or DNS
IN	ResetToFactoryDefaults	boolean	If set to “true”, all configuration information is replaced with the auto discovery factory defaults. If set to “false”, an error will be returned
IN	PerformAutoDiscovery	uint16	A value of "Off = 1" disables auto discovery. A value of "Now = 2" enables and initiates auto discovery immediately. A value of “NextBoot = 3" will delay reconfiguration and auto discovery until next powercycle.
OUT	MessageID	string	Error Message ID- can be used to index into Dell Message registry files
OUT	Message	string	Error Message in English corresponding to MessageID is returned if the method fails to execute
OUT	MessageArguments	string[]	Substitution variables for dynamic error messages

8.5 DCIM_LCService.ClearProvisioningServer()

A method used to clear the provisioning server values.

Table 12 – DCIM_LCService.ClearProvisioningServer() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred

Table 13 – DCIM_LCService.ClearProvisioningServer() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT (optional)	MessageID	string	Error MessageID is returned if the method fails to execute
OUT (optional)	Message	string	Error Message in English corresponding to MessageID is returned if the method fails to execute

8.6 DCIM_LCService.DownloadServerPublicKey()

A method used to download the server public key to the LC.

Table 14 – DCIM_LCService.DownloadServerPublicKey() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

Table 15 – DCIM_LCService.DownloadServerPublicKey() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, OctetString	KeyContent	string	Base64 encoded public key content.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of public key download status
OUT	MessageID	string	Error Message ID- can be used to index into Dell Message registry files
OUT	Message	string	Error Message in English corresponding to MessageID is returned if the method fails to execute
OUT	MessageArguments	string[]	Substitution variables for dynamic error messages

8.7 DCIM_LCService.DownloadClientCerts()

A method used to download the client private certificate, password, and root certificate to LC.

Table 16 – DCIM_LCService.DownloadClientCerts() Method: Return Code Values

Value	Description
0	Request was successfully executed.
1	Method is unsupported.
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

Table 17 – DCIM_LCService.DownloadClientCerts() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, OctetString	KeyContent	string	Base64 encoded private key content.
IN	Password	string	Private key password
IN, OctetString	CAConcent	string	Base64 encoded root certificate content
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of public key download status
OUT (optional)	MessageID	string	Error Message ID- can be used to index into Dell Message registry files
OUT	Message	string	Error Message in English corresponding to MessageID is returned if the method fails to execute
OUT	MessageArguments	string[]	Substitution variables for dynamic error messages

9 Use Cases

This section contains object diagrams and use cases for the LC Management Profile.

9.1 Object Diagrams

Figure 2 represents a possible instantiation of the LC Management Profile. In this instantiation, the ComputerSystem is not represented.

For simplicity, the prefix CIM_ has been removed from the names of the standard classes in the figure.

9.2 Use Case 1 – Representing CCR, System ID and DHS attributes

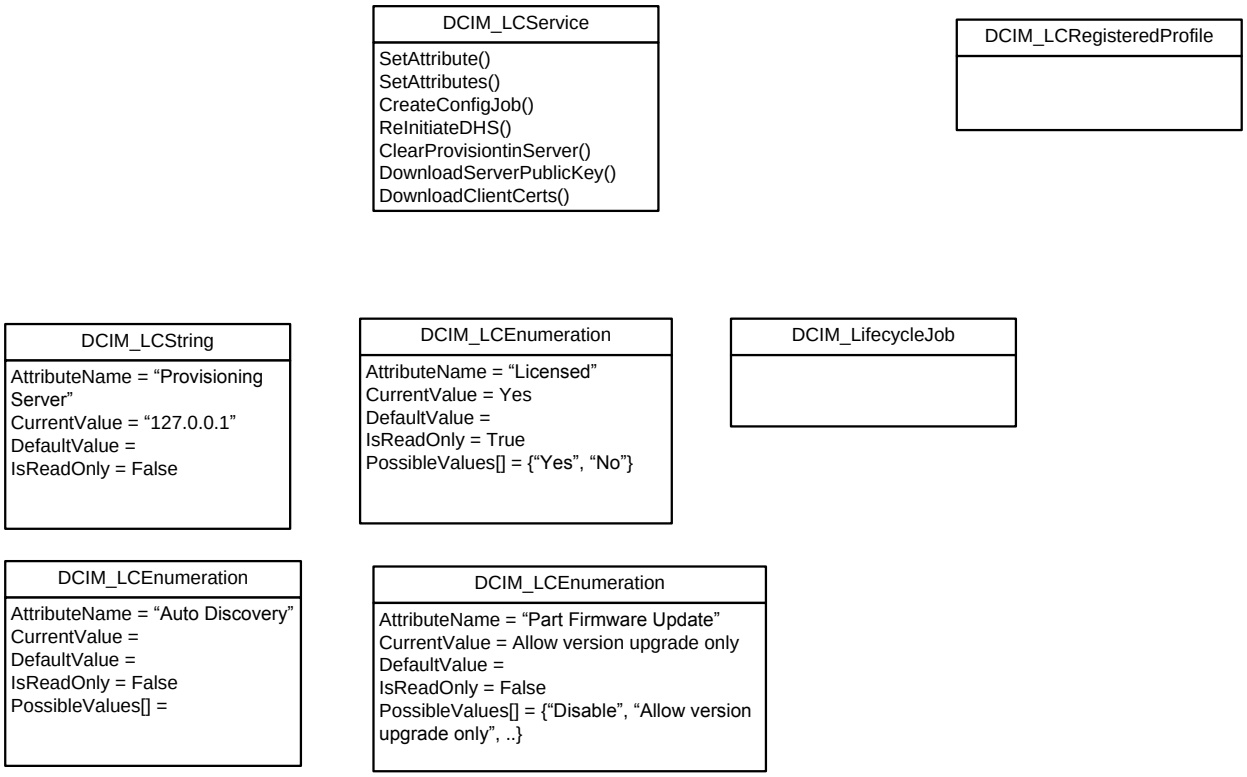

Figure 2 – Object Diagram representing CCR, System ID and DHS attributes

10 CIM Elements

Table shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be implemented as described in Table . Section 0 "Implementation Requirements" and "Methods" may impose additional requirements on these elements.

Table 18 – CIM Elements: iDRAC Attribute Profile

Element Name	Requirement	Description
Classes		
DCIM_LCAttribute	Mandatory	See subclause 10.1
DCIM_LCService	Mandatory	See subclause 10.2
DCIM_LCEnumeration	Mandatory	See subclause 10.3
DCIM_LCString	Mandatory	See subclause 10.4
DCIM_LCRegisteredProfile	Mandatory	See subclause 10.5
Indications		
None defined in this profile		

10.1 DCIM_LCAttribute

DCIM_LCAttribute is a trivial derivation of CIM_BIOSAttribute. The CIM_BIOSAttribute class is implemented to represent all LC attributes. Table21 contains the element requirements for this class.

Table 19 – Class: DCIM_LCAttribute

Properties	Notes	Description
InstanceID	Mandatory	**Key:** This element shall specify the unique identifier for an instance of this class within the Implementation namespace.
AttributeName	Mandatory	
CurrentValue	Mandatory	Current value of this attribute
DefaultValue	Optional	Contains the default value for this attribute
PendingValue	Conditional	Pending value of this attribute set via DCIM_LCService.SetAttribute or SetAttributes methods.
IsReadOnly	Mandatory	True if current value cannot be changed

10.2 DCIM_LCService

DCIM_LCService is used to provide a central class for the LC Management Profile, and to provide a place for the SetAttribute() and SetAttributes() methods that will change LC configuration attribute values.

Table 20 – Class: DCIM_LCService

Properties and Methods	Requirement	Description
SystemCreationClassName	Mandatory	**Key**
CreationClassName	Mandatory	**Key**
SystemName	Mandatory	**Key**
Name	Mandatory	**Key**
ElementName	Mandatory	This property shall have value of "LC Service".

10.3 DCIM_LCEnumeration

DCIM_LCEnumeration is derived from DCIM_LCAttribute, and also contains properties from CIM_BIOSEnumeration.

Table 21 – Class: DCIM_LCEnumeration

Properties and Methods	Requirement	Description
Inherited from DCIM_LCAttribue	Mandatory	Keys and mandatory properties inherited from DCIM_LCAttribute
PossibleValues[ ]	Mandatory	

10.4 DCIM_LCString

DCIM_LCString is derived from DCIM_LCAttribute, and also contains properties from CIM_BIOSString.

Table 22 – Class: DCIM_LCString

Properties and Methods	Requirement	Description
Inherited from DCIM_LCAttribue	Mandatory	Keys and mandatory properties inherited from DCIM_LCAttribute
StringType	Mandatory	

10.5 DCIM_LCRegisteredProfile

DCIM_LCRegisteredProfile is a trivial derivation of CIM_RegisteredProfile.The CIM_RegisteredProfile class is defined by the Profile Registration Profile. The requirements denoted in 25 are in addition to those mandated by the Profile Registration Profile.

Table 23 – Class: CIM_RegisteredProfile

Properties	Requirement	Description
RegisteredName	Mandatory	This property shall have a value of “Lifecycle Controller Management”.
RegisteredVersion	Mandatory	This property shall have a value of “1.0.0”.
RegisteredOrganization	Mandatory	This property shall have a value of 1 (Other).
OtherRegisteredOrganization	Mandatory	This property shall match “DCIM”

ANNEX A
(informative)

Dell Extension MOF

```
// Copyright (c) 2009 Dell Inc.  All rights reserved.
// ==================================================================
// DCIM_LCEnumeration
// ==================================================================
   [Version ( "1.0.0" ), Description (
       "This class extends the DCIM_LCAttribute base class definition "
       "to provide informational detail for enumeration data types. "
       "Enumerations are single-selection lists with a finite set of "
       "entities. Example enumerations might include \"Power-On Self "
       "Test: Enable, Disable\" or \"Standby Timeout: 15 minutes, 30 "
       "minutes, 45 minutes, Never\"." )]
class DCIM_LCEnumeration : DCIM_LCAttribute {

      [Required, Description (
          "An array of strings containing possible value "
          "definitions for the Attribute. The first element SHOULD "
          "not be NULL and there SHOULD be no NULL or empty string "
          "elements intermixed with populated array element values. "
          "A Value of NULL SHALL indicate an element\'s value is "
          "unknown." ),
       ArrayType ( "indexed" ),
       ModelCorrespondence {
          "DCIM_LCEnumeration.PossibleValuesDescription" }]
   string PossibleValues[];

      [Description (
          "An array of free-form strings providing explanations and "
          "details behind the entries in the "
          "DCIM_LCEnumeration.PossibleValues array. Note that each "
          "entry of this array is related to the entry in "
          "PossibleValues array that is located at the same index." ),
       ArrayType ( "indexed" ),
       ModelCorrespondence { "DCIM_LCEnumeration.PossibleValues" }]
   string PossibleValuesDescription[];
};


// Copyright (c) 2009 Dell Inc.  All rights reserved.
// ==================================================================
// DCIM_LCAttribute
// ==================================================================
  [Version ( "1.0.0" ), Description(
       "DCIM_LCAttribute is a trivial derivation of "
       "CIM_BIOSAttribute.")]
class DCIM_LCAttribute : CIM_BIOSAttribute
{
};
```

```
// Copyright (c) 2009 Dell Inc.  All rights reserved.
// ==================================================================
// DCIM_LCString
// ==================================================================
  [Version ( "1.0.0" ),Description(
       "The DCIM_LCString object may be used to instantiate "
       "string-based LC attributes.")]
class DCIM_LCString : DCIM_LCAttribute {
      [Description (
          "The StringType property provides management applications "
          "with information defining implementation\'s capabilities "
          "regarding string values. A value of ASCII indicates a "
          "plain text formatting with no tabs formating or bolds, "
          "plain text. A value of hex indicates a simple encoding "
          "of the numerals 0 to 9 and the letters A to F and is not "
          "case sensitive. A value of unicode indicates fully "
          "multi-lingual text support. A string LC Attribute with "
          "value of regex is a regular expression and the "
          "DCIM_LCString.ValueExpression shall be applied to "
          "values represented and any value that is attempted to be "
          "set by invoking SetAttribute()." ),
       ValueMap { "0", "2", "3", "4", "5", "30..65535",
          "65536..4294967295" },
       Values { "Unknown", "ASCII", "hex", "unicode", "regex",
          "DMTF-Reserved", "Vendor-Specific" }]
   uint32 StringType;

      [Description (
          "This property specifies the minimum string length "
          "allowed when modifying this BIOS attribute. A value of "
          "NULL SHALL be defined as zero." ),
       ModelCorrespondence { "DCIM_LCString.MaxLength" }]
   uint64 MinLength;

      [Description (
          "This property specifies the maximum string length "
          "allowed when modifying this LC attribute. A value of "
          "NULL defines the maximum length is unknown. A value of 0 "
          "defines the maximum length as 18446744073709551615, the "
          "limitation of the property definition as uint64." ),
       ModelCorrespondence { "DCIM_LCString.MinLength" }]
   uint64 MaxLength;

      [Description (
          "The property denotes a Perl-compatible regular "
          "expression (PCRE) syntax to use in validating Attribute "
          "values. For a string Attribute where StringType=\"regex\" "
          "this property shall have a value." )]
   string ValueExpression;
};
```

```
// Copyright (c) 2009 Dell Inc.  All rights reserved.
// ==================================================================
// DCIM_LCService
// ==================================================================
  [Version ( "1.0.0" ), Description(
       "DCIM_LCService is a derivation of CIM_BIOSService "
       "with extrinsic method extensions to support "
       "Lifecycle Controller functionality)"]
class DCIM_LCService : CIM_BIOSService {
      [Experimental, Description (
          "This method is called to modify instances of "
          "LCAttribute associated with this LCService." ),
       ValueMap { "0", "1", "2", "3..32767", "32768..65535" },
       Values { "Completed with No Error", "Not Supported", "Error",
          "DMTF Reserved", "Vendor Specified" }]
   uint32 SetAttribute(
         [Required, IN, Description (
             "A reference to a LCAttribute specified by "
             "AttributeName of the LCAttribute to be modified." )]
      string AttributeName,
         [Required, IN, Description (
             "A new value to assign to the specified "
             "LCAttribute. If this value is valid, it will be "
             "applied to the CurrentValue or PendingValue "
             "property of the specified LCAttribute depending "
             "on the Lifecycle Controller implementation and any "
             "requirements for a system restart." )]
      string AttributeValue[],
         [OUT, Description (
             "Invoking the SetAttribute may result in a set "
             "to CurrentValue property or the PendingValue "
             "property. SetResult returns the result of invoking "
             "the SetAttribute method when the method return "
             "code is 0 completed with no error." ),
          ValueMap { "2", "3", "4..32767", "32768..65535" },
          Values { "Set CurrentValue property",
             "Set PendingValue property", "DMTF Reserved",
             "Vendor Specified" }]
      string SetResult,
         [OUT, Description (
             "Invoking the SetAttribute may require "
             "a reboot for the set to be effective. "
             "Value \"Yes\" means a reboot is required. "
             "Value \"No\" means a reboot is not required")]
      string RebootRequired,
         [OUT, Description (
             "Error MessageID is returned if the method fails "
             "to execute." )]
      string MessageID,
         [OUT, Description (
```

```
            "Error Message in english corresponding to the "
            "MessageID" )]
      string Message,
         [OUT, Description (
            "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);

   uint32 SetAttributes(
         [Required, IN, Description (
            "A reference to a LCAttribute specified by "
            "AttributeName of the LCAttribute to be modified." )]
      string AttributeName[],
         [Required, IN, Description (
            "A new value to assign to the specified "
            "LCAttribute. If this value is valid, it will be "
            "applied to the CurrentValue or PendingValue "
            "property of the specified LCAttribute depending "
            "on the Lifecycle Controller implementation and any "
            "requirements for a system restart." )]
      string AttributeValue[],
         [OUT, Description (
            "Invoking the SetAttributes may result in a set "
            "to CurrentValue property or the PendingValue "
            "property. SetResult returns the result of invoking "
            "the SetAttributes method when the method return "
            "code is 0 completed with no error." ),
          ValueMap { "2", "3", "4..32767", "32768..65535" },
          Values { "Set CurrentValue property",
            "Set PendingValue property", "DMTF Reserved",
            "Vendor Specified" }]
      string SetResult[],
         [OUT, Description (
            "Invoking the SetAttribute may require "
            "a reboot for the set to be effective. "
            "Value \"Yes\" means a reboot is required. "
            "Value \"No\" means a reboot is not required")]
      string RebootRequired[],
         [OUT, Description (
            "Error MessageID is returned if the method fails "
            "to execute." )]
      string MessageID,
         [OUT, Description (
            "Error Message in english corresponding to the "
            "MessageID" )]
      string Message,
         [OUT, Description (
            "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);

      [Experimental, Description (
          "This method is called to reinitialize the Lifecycle "
          "Controller to factory default settings with "
          "auto discovery enabled." ),
       ValueMap { "0", "1", "2", "3..32767", "32768..65535" },
       Values { "Completed with No Error", "Not Supported", "Error",
          "DMTF Reserved", "Vendor Specified" }]
```

```
   uint32 ReInitiateDHS(
         [IN, Description (
             "This is an optional parameter to specify "
             "Provisioning Server addresses and ports "
             "used for auto discovery, if omitted, the "
             "Lifecycle Controller will get the value "
             "from DHCP or DNS" )]
      string ProvisioningServer,
         [Required, IN, Description (
             "If true, all configuration information is "
             "replaced with auto discovery factory defaults. "
             "If false, an error will be returned" )]
      boolean ResetToFactoryDefaults,
         [Required, IN ( false ), Description (
             "\"Off\" disables auto discovery, "
             "\"Now\" enables and initiates auto discovery "
             "immediately and "
             "\"NextBoot\" will delay reconfiguration and "
             "auto discovery until next powercycle." ),
          ValueMap { "1", "2", "3", "4..32767", "32768..65535" },
          Values { "Off", "Now", "NextBoot", "DMTF Reserved", "Vendor
Specified" }]
      uint16 PerformAutoDiscovery,
         [OUT, Description (
             "Error MessageID is returned if the method fails "
             "to execute." )]
      string MessageID,
         [OUT, Description (
             "Error Message in english corresponding to the "
             "MessageID" )]
      string Message,
         [OUT, Description (
             "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);

      [Experimental, Description (
          "This method is called to apply the pending values set "
          "using the SetAttribute and SetAttributes methods" ),
       ValueMap { "0", "1", "2", "3..32767", "32768..65535" },
       Values { "Completed with No Error", "Not Supported", "Error",
          "DMTF Reserved", "Vendor Specified" }]
   uint32 CreateConfigJob(
         [IN, Description (
             "Scheduled start time for the job." )]
      datetime ScheduledStartTime,
         [IN, Description (
             "a value of \"true\" will reboot the system if "
             "reboot is required "
             "a value of \"false\" will not reboot the system "
             "if reboot is required" )]
      boolean RebootIfRequired,
         [OUT, Description (
             "a reference to the ConcreteJob is returned" )]
      CIM_ConcreteJob REF Job,
         [OUT, Description (
             "Error MessageID is returned if the method fails "
```

```
            "to execute." )]
      string MessageID,
         [OUT, Description (
            "Error Message in english corresponding to the "
            "MessageID" )]
      string Message,
         [OUT, Description (
            "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);

      [Experimental, Description (
          "This method is called to clear the values for "
          "provisioning server" ),
       ValueMap { "0", "1", "2", "3..32767", "32768..65535" },
       Values { "Completed with No Error", "Not Supported", "Error",
          "DMTF Reserved", "Vendor Specified" }]
   uint32 ClearProvisioningServer(
         [OUT, Description (
            "Error MessageID is returned if the method fails "
            "to execute." )]
      string MessageID,
         [OUT, Description (
            "Error Message in english corresponding to the "
            "MessageID" )]
      string Message,
         [OUT, Description (
            "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);

      [Experimental, Description (
          "This method is called to replace the CA certificate "
          "used to authenticate the auto discovery provisioning server" ),
       ValueMap { "0", "1", "2", "3..32767", "32768..65535" },
       Values { "Completed with No Error", "Not Supported", "Error",
          "DMTF Reserved", "Vendor Specified" }]
   uint32 DownloadServerPublicKey(
         [IN, Description (
            "base64 encoded string containing the "
            "CA certificate content" )]
      string KeyContent,
         [OUT, Description (
            "a reference to the ConcreteJob is returned" )]
      CIM_ConcreteJob REF Job,
         [OUT, Description (
            "Error MessageID is returned if the method fails "
            "to execute." )]
      string MessageID,
         [OUT, Description (
            "Error Message in english corresponding to the "
            "MessageID" )]
      string Message,
         [OUT, Description (
            "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);

      [Experimental, Description (
```

```
          "This method is called to generate a client certificate "
          "and private key using the provided CA certificate, "
          "CA private key and CA password" ),
       ValueMap { "0", "1", "2", "3..32767", "32768..65535" },
       Values { "Completed with No Error", "Not Supported", "Error",
          "DMTF Reserved", "Vendor Specified" }]
   uint32 DownloadClientCerts(
         [IN, Description (
             "base64 encoded string containing "
             "the CA private key content" )]
      string KeyContent,
         [IN, Description (
             "CA private key password" )]
      string Password,
         [IN, Description (
             "base64 encoded string containing "
             "CA Certificate content" )]
      string CAContent,
         [OUT, Description (
             "a reference to the ConcreteJob is returned" )]
      CIM_ConcreteJob REF Job,
         [OUT, Description (
             "Error MessageID is returned if the method fails "
             "to execute." )]
      string MessageID,
         [OUT, Description (
             "Error Message in english corresponding to the "
             "MessageID" )]
      string Message,
         [OUT, Description (
             "Any dynamic string substitutions for the Message" )]
      string MessageArguments[]);
};
```