

1 **Lifecycle Controller**
2 **(LC) Management Profile**

3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29

Document Number: DCIM1039
Document Type: Specification
Document Status: Published
Document Language: E
Date: 2012-12-30

Version: 1.6.0



30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51

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

58

59 © 2011 - 2012 Dell Inc. All rights reserved. Reproduction in any manner whatsoever without the express
60 written permission of Dell, Inc. is strictly forbidden. For more information, contact Dell.

61

62 *Dell* and the *DELL* logo are trademarks of Dell Inc. *Microsoft* and *WinRM* are either trademarks or
63 registered trademarks of Microsoft Corporation in the United States and/or other countries. Other

64 trademarks and trade names may be used in this document to refer to either the entities claiming the
65 marks and names or their products. Dell disclaims proprietary interest in the marks and names of others.
66

CONTENTS

67			
68	1	Scope	8
69	2	Normative References.....	8
70	3	Terms and Definitions	8
71	4	Symbols and Abbreviated Terms	10
72	5	Synopsis	10
73	6	Description	11
74	7	Implementation Requirements	12
75	7.1	DCIM_LCService	13
76	7.2	DCIM_LCEnumeration.....	14
77	7.3	DCIM_LCString.....	16
78	7.4	Lifecycle Controller (LC) Management Registration – DCIM_LCRegistered Profile	19
79	8	Methods.....	20
80	8.1	DCIM_LCService.SetAttribute().....	21
81	8.2	DCIM_LCService.SetAttributes()	22
82	8.3	DCIM_LCService.CreateConfigJob().....	24
83	8.4	DCIM_LCService.ReInitiateDHS()	25
84	8.5	DCIM_LCService.ReInitiateAutoDiscovery()	27
85	8.6	DCIM_LCService.ClearProvisioningServer().....	30
86	8.7	DCIM_LCService.DownloadServerPublicKey()	30
87	8.8	DCIM_LCService.DownloadClientCerts().....	31
88	8.9	DCIM_LCService.DeleteAutoDiscoveryClientCerts ()	33
89	8.10	DCIM_LCService.SetCertificateAndPrivateKey()	35
90	8.11	DCIM_LCService.SetPublicCertificate()	36
91	8.12	DCIM_LCService.DeleteAutoDiscoveryServerPublicKey()	36
92	8.13	DCIM_LCService.InsertCommentInLCLog()	37
93	8.14	DCIM_LCService.ExportLCLog().....	38
94	8.15	DCIM_LCService.ExportHWInventory()	39
95	8.16	DCIM_LCService.ExportFactoryConfiguration()	41
96	8.17	DCIM_LCService.ExportSystemConfiguration()	43
97	8.18	DCIM_LCService.ImportSystemConfiguration()	45
98	8.19	DCIM_LCService.LCWipe().....	46
99	8.20	DCIM_LCService.TestNetworkShare().....	47
100	8.21	DCIM_LCService.BackupImage().....	48
101	8.22	DCIM_LCService.RestoreImage()	50
102	8.23	DCIM_LCService.GetRSStatus().....	52
103	8.24	DCIM_LCService.GetRemoteServicesAPIStatus ()	53
104	8.25	DCIM_LCService.ExportCertificate()	54
105	9	Use Cases	56
106	10	CIM Elements.....	56
107	11	Privilege and License Requirement	56
108		ANNEX A (informative) Change Log.....	60
109			

111	Figures	
112	Figure 1 – LC Management Profile: Class Diagram	12
113		

114	Tables	
115	Table 1 – Related Profiles	11
116	Table 2 – Class Requirements: Power State Management Profile.....	12
117	Table 3 – DCIM_LCService - Operations	13
118	Table 4 – DCIM_LCService - Properties.....	13
119	Table 2 – DCIM_LCEnumeration - Operations.....	14
120	Table 3 – Class: DCIM_LCEnumeration	15
121	Table 4 – DCIM_LCEnumeration Attributes.....	16
122	Table 5 – DCIM_LCString - Operations	17
123	Table 6 – Class: DCIM_LCString	18
124	Table 7 – DCIM_LCString Attributes.....	18
125	Table 8 – DCIM_LCRegisteredProfile - Operations.....	19
126	Table 9 – Class: DCIM_LCRegisteredProfile.....	20
127	Table 10 – DCIM_LCService.SetAttribute() Method: Return Code Values	21
128	Table 11 – DCIM_LCService.SetAttribute() Method: Parameters	21
129	Table 12 – DCIM_LCService.SetAttribute() Method: Standard Messages.....	22
130	Table 13 – DCIM_LCService.SetAttributes() Method: Return Code Values	23
131	Table 14 – DCIM_LCService.SetAttributes() Method: Parameters	23
132	Table 15 – DCIM_LCService.SetAttributes() Method: Standard Messages	24
133	Table 16 – DCIM_LCService.CreateConfigJob() Method: Return Code Values	25
134	Table 17 – DCIM_LCService.CreateConfigJob() Method: Parameters	25
135	Table 18 – DCIM_LCService.CreateConfigJob() Method: Standard Messages	25
136	Table 19 – DCIM_LCService.ReInitiateDHS() Method: Return Code Values	26
137	Table 20 – DCIM_LCService.ReInitiateDHS() Method: Parameters	26
138	Table 21 – DCIM_LCService.ReInitiateDHS() Method: Standard Messages.....	26
139	Table 19 – DCIM_LCService.ReInitiateAutoDiscovery() Method: Return Code Values.....	28
140	Table 20 – DCIM_LCService.ReInitiateAutoDiscovery() Method: Parameters	28
141	Table 21 – DCIM_LCService.ReInitiateAutoDiscovery() Method: Standard Messages	29
142	Table 22 – DCIM_LCService.ClearProvisioningServer() Method: Return Code Values	30
143	Table 23 – DCIM_LCService.ClearProvisioningServer() Method: Parameters	30
144	Table 24 – DCIM_LCService.ClearProvisioningServer() Method: Standard Messages.....	30
145	Table 25 – DCIM_LCService.DownloadServerPublicKey() Method: Return Code Values	31
146	Table 26 – DCIM_LCService.DownloadServerPublicKey() Method: Parameters	31
147	Table 27 – DCIM_LCService.DownloadServerPublicKey() Method: Standard Messages.....	31
148	Table 28 – DCIM_LCService.DownloadClientCerts() Method: Return Code Values	33
149	Table 29 – DCIM_LCService.DownloadClientCerts() Method: Parameters	33
150	Table 30 – DCIM_LCService.DownloadClientCerts() Method: Standard Messages.....	33
151	Table 31 – DCIM_LCService.DeleteAutoDiscoveryClientCerts() Method: Return Code Values	34
152	Table 32 – DCIM_LCService.DeleteAutoDiscoveryClientCerts() Method: Parameters	34
153	Table 33 – DCIM_LCService.DeleteAutoDiscoveryClientCerts() Method: Standard Messages.....	34
154	Table 34 – DCIM_LCService.SetCertificateAndPrivateKey() Method: Return Code Values	35
155	Table 35 – DCIM_LCService.SetCertificateAndPrivateKey() Method: Parameters	35

156	Table 36 – DCIM_LCService.SetCertificateAndPrivateKey() Method: Standard Messages	35
157	Table 37 – DCIM_LCService.SetPublicCertificate() Method: Return Code Values	36
158	Table 38 – DCIM_LCService.SetPublicCertificate() Method: Parameters	36
159	Table 39 – DCIM_LCService.SetPublicCertificate() Method: Standard Messages	36
160	Table 40 – DCIM_LCService.DeleteAutoDiscoveryServerPublicKey() Method: Return Code Values.....	37
161	Table 41 – DCIM_LCService.DeleteAutoDiscoveryServerPublicKey() Method: Parameters	37
162	Table 42 – DCIM_LCService.DeleteAutoDiscoveryServerPublicKey() Method: Standard Messages	37
163	Table 43 – DCIM_LCService.InsertCommentInLCLog() Method: Return Code Values.....	37
164	Table 44 – DCIM_LCService.InsertCommentInLCLog() Method: Parameters.....	37
165	Table 45 – DCIM_LCService.InsertCommentInLCLog() Method: Standard Messages	38
166	Table 46 – DCIM_LCService.ExportLCLog() Method: Return Code Values	38
167	Table 47 – DCIM_LCService.ExportLCLog() Method: Parameters	38
168	Table 48 – DCIM_LCService.ExportLCLog() Method: Standard Messages	39
169	Table 49 – DCIM_LCService.ExportHWInventory() Method: Return Code Values	40
170	Table 50 – DCIM_LCService.ExportHWInventory() Method: Parameters.....	40
171	Table 51 – DCIM_LCService.ExportHWInventory() Method: Standard Messages	40
172	Table 52 – DCIM_LCService.ExportFactoryConfiguration() Method: Return Code Values	42
173	Table 53 – DCIM_LCService.ExportFactoryConfiguration() Method: Parameters	42
174	Table 54 – DCIM_LCService.ExportFactoryConfiguration() Method: Standard Messages.....	42
175	Table 55 – DCIM_LCService.ExportSystemConfiguration() Method: Return Code Values	44
176	Table 56 – DCIM_LCService.ExportSystemConfiguration() Method: Parameters	44
177	Table 57 – DCIM_LCService.ExportSystemConfiguration() Method: Standard Messages.....	44
178	Table 56 – DCIM_LCService.ImportSystemConfiguration() Method: Parameters	45
179	Table 57 – DCIM_LCService.ImportSystemConfiguration() Method: Standard Messages.....	45
180	Table 58 – DCIM_LCService.LCWipe() Method: Return Code Values	47
181	Table 59 – DCIM_LCService.LCWipe() Method: Parameters	47
182	Table 60 – DCIM_LCService.LCWipe() Method: Standard Messages.....	47
183	Table 62 – DCIM_LCService.TestNetworkShare() Method: Parameters	48
184	Table 63 – DCIM_LCService.TestNetworkShare() Method: Standard Messages.....	48
185	Table 62 – DCIM_LCService.BackupImage() Method: Parameters	49
186	Table 63 – DCIM_LCService.BackupImage() Method: Standard Messages	50
187	Table 64 – DCIM_LCService.RestoreImage() Method: Return Code Values	51
188	Table 65 – DCIM_LCService.RestoreImage() Method: Parameters	51
189	Table 66 – DCIM_LCService.RestoreImage() Method: Standard Messages	52
190	Table 67 – DCIM_LCService.GetRSStatus() Method: Return Code Values	53
191	Table 68 – DCIM_LCService.GetRSStatus() Method: Parameters	53
192	Table 69 – DCIM_LCService.GetRSStatus() Method: Standard Messages	53
193	Table 67 – DCIM_LCService.GetRemoteServicesAPIStatus() Method: Return Code Values.....	54
194	Table 68 – DCIM_LCService.GetRemoteServicesAPIStatus() Method: Parameters.....	54
195	Table 70 – DCIM_LCService.GetRemoteServicesAPIStatus() Method: Standard Messages	54
196	Table 71 – DCIM_LCService.ExportCertificate() Method: Return Code Values	55
197	Table 72 – DCIM_LCService.ExportCertificate() Method: Parameters	55
198	Table 73 – DCIM_LCService.ExportCertificate() Method: Standard Messages.....	55
199	Table 74 – Privilege and License Requirements	56

201

Lifecycle Controller (LC) Management

202 1 Scope

203 The Lifecycle Controller (LC) Management Profile describes the management of the Dell Lifecycle
204 Controller and its configuration attributes. The profile details certificate management and the LC
205 configuration services that are necessary for the LC's proper functionality. The LC configuration attributes
206 are modeled as attribute collections for an individual LC; typically there is one LC per system platform.

207 2 Normative References

208 Refer to the following documents for more information.

209 **NOTE:** For dated references, only the edition cited applies. For undated references, the latest edition of
210 the referenced document (including any amendments) applies.

- 211 • DMTF DSP1033, *Profile Registration Profile 1.0*
- 212 • DMTF DSP1061, *BIOS Management Profile 1.0*
- 213 • *Dell Job Control Profile 1.0.0*
- 214 • DMTF DSP0004, *CIM Infrastructure Specification 2.3*
- 215 • *Dell Lifecycle Controller Best Practices Guide 1.0*,
- 216 http://en.community.dell.com/techcenter/extras/m/white_papers/20066173.aspx
- 217 • *Dell WSMAN Licenses and Privileges 1.0*
- 218 • ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
- 219 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>
- 220 • Unified Modeling Language (UML) from the Open Management Group (OMG),
- 221 <http://www.uml.org>
- 222 • Dell Tech Center MOF Library: <http://www.delltechcenter.com/page/DCIM.Library.MOF>
- 223 • Related Managed Object Format (MOF) files:
 - 224 ○ DCIM_LCAAttribute.mof
 - 225 ○ DCIM_LCElementConformsToProfile.mof
 - 226 ○ DCIM_LCEnumeration.mof
 - 227 ○ DCIM_LCRegisteredProfile.mof
 - 228 ○ DCIM_LCService.mof
 - 229 ○ DCIM_LCString.mof

230 3 Terms and Definitions

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

- 232 **3.1**
233 **Conditional** – Indicates requirements to be followed strictly in order to conform to the document when the
234 specified conditions are met.
- 235 **3.2**
236 **Mandatory** – Indicates requirements to be followed strictly in order to conform to the document and from
237 which no deviation is permitted.
- 238 **3.3**
239 **May** – Indicates a course of action permissible within the limits of the document.
- 240 **3.4**
241 **Optional** – Indicates a course of action permissible within the limits of the document.
- 242 **3.5**
243 **can** – Used for statements of possibility and capability, whether material, physical, or causal.
- 244 **3.6**
245 **cannot** – Used for statements of possibility and capability, whether material, physical, or causal.
- 246 **3.7**
247 **need not** – Indicates a course of action permissible within the limits of the document.
- 248 **3.8**
249 **referencing profile** – Indicates a profile that owns the definition of this class and can include a reference
250 to this profile in its “Related Profiles” table.
- 251 **3.9**
252 **shall** – Indicates requirements to be followed strictly in order to conform to the document and from which
253 no deviation is permitted.

- 254 **3.10**
255 **shall not** – Indicates requirements to be followed strictly in order to conform to the document and from
256 which no deviation is permitted.
- 257 **3.11**
258 **should** – Indicates that among several possibilities, one is recommended as particularly suitable, without
259 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required.
- 260 **3.12**
261 **should not** – Indicates that a certain possibility or course of action is deprecated but not
262 prohibited
- 263 **3.13**
264 **FQDD** – Fully Qualified Device Descriptor is used to identify a particular component in a system.
- 265 **3.14**
266 **Interop Namespace** – Interop Namespace is where instrumentation instantiates classes to advertise its
267 capabilities for client discovery.
- 268 **3.15**
269 **Implementation Namespace** – Implementation Namespace is where instrumentation instantiates
270 classes relevant to executing core management tasks.
- 271 **3.16**
272 **ENUMERATE** – Refers to WS-MAN `ENUMERATE` operation as described in Section 8.2 of
273 `DSP0226_V1.1` and Section 9.1 of `DSP0227_V1.0`
- 274 **3.17**
275 **GET** – Refers to WS-MAN `GET` operation as defined in Section 7.3 of `DSP00226_V1.1` and Section 7.1
276 of `DSP0227_V1.0`

277 **4 Symbols and Abbreviated Terms**

278 **4.1**
279 **CIM** – Common Information Model

280 **4.2**
281 **LC** – Lifecycle Controller

282 **4.3**
283 **VD** – Virtual Disk

284 **5 Synopsis**

285 **Profile Name:** LC Management

286 **Version:** 1.6.0

287 **Organization:** DCIM

288 **CIM Schema Version:** 2.26 Experimental

289 **Central Class:** DCIM_LCService

290 **Scoping Class:** CIM_ComputerSystem

291 The LC Management Profile extends the management capability of the referencing profiles by adding the
292 capability to describe lifecycle controller (LC) through its attributes. Each DCIM_LCAttribute derived
293 instance represents an LC configuration-related attribute.

294 The DCIM_LCService class shall be the Central Class. The DCIM_ComputerSystem class shall be the
295 Scoping Class. The DCIM_LCService instance shall be the Central Instance. The
296 DCIM_ComputerSystem instance shall be the Scoping Instance.

297 Table 1 identifies profiles that are related to this profile.

298

Table 1 – Related Profiles

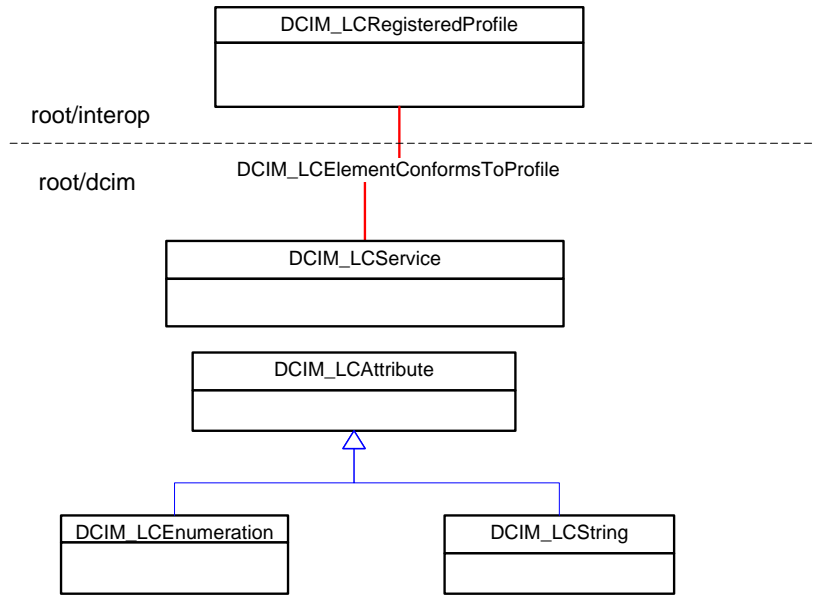
Profile Name	Organization	Version	Relationship
Profile Registration	DCIM	1.0	Reference

299 **6 Description**

300 The LC Management Profile describes the LC attribute configuration service and the attributes instances
301 that the service manages. The profile also describes the relationship of the LC attribute service to the Dell
302 profile version information.

303 Figure 1 represents the class schema for the Lifecycle Controller (LC) Management. The LC service in a
304 managed system is represented by the instance of the DCIM_LCService class. The LC attributes are
305 represented by the DCIM_LCAttribute class derivation: DCIM_LCEnumeration and DCIM_LCString
306 classes.

307 The LC Management Profile information is represented with the instance of the
308 DCIM_LCRegisteredProfile.



309

310

Figure 1 – LC Management Profile: Class Diagram

311 7 Implementation Requirements

312 This section describes the implementation of Dell LC Management.

313

Table 2 – Class Requirements: Power State Management Profile

Element Name	Requirement	Description
Classes		
DCIM_LCService	Mandatory	The class shall be implemented in the <i>Implementation Namespace</i> . See section 7.1.
DCIM_LCEnumeration	Mandatory	The class shall be implemented in the <i>Implementation Namespace</i> . See section 7.2.
DCIM_LCString	Mandatory	The class shall be implemented in the <i>Implementation Namespace</i> . See section 7.3.
DCIM_LCElementConformsToProfile	Mandatory	The class shall be implemented in both the <i>Interop</i> and <i>Implementation Namespaces</i> . See section 7.1 and 7.4.
DCIM_LCRegisteredProfile	Mandatory	The class shall be implemented in the <i>Interop Namespace</i> . See section 7.4.
Indications		
None defined in this profile		

314 7.1 DCIM_LCService

315 This section describes the implementation for the DCIM_LCService class.

316 This class is instantiated in the Implementation Namespace.

317 The DCIM_LCElementConformsToProfile association's ManagedElement property shall references the
318 DCIM_LCService instance(s).

319 7.1.1 Resource URIs for WinRM®

320 The class Resource URI is:

321 *"http://schemas.dell.com/wbem/wscim/1/cim-*
322 *schema/2/DCIM_LCService?__cimnamespace=root/dcim"*

323 The key properties are the SystemCreationClassName, CreationClassName, SystemName, Name

324 The instance Resource URI for DCIM_LCService instance is:

325 *"http://schemas.dell.com/wbem/wscim/1/cim-*
326 *schema/2/DCIM_LCService?__cimnamespace=root/dcim+SystemCreationClassName=DCIM_Com*
327 *puterSystem+SystemName=DCIM:ComputerSystem+CreationClassName=DCIM_LCService+Name*
328 *=DCIM:LCService"*

329 7.1.2 Operations

330 The following table lists the operations implemented on DCIM_LCService.

331 **Table 3 – DCIM_LCService - Operations**

Operation Name	Requirements	Required Input
Get	Mandatory	Instance URI
Enumerate	Mandatory	Class URI
Invoke	Mandatory	Instance URI and Method parameters

332

333 7.1.3 Class Properties

334 The following table lists the implemented properties for DCIM_LCService instance in a system. The
335 "Requirements" column shall denote whether the property is implemented (for requirement definitions,
336 see section 3.1, 3.2, and 3.4). The "Additional Requirements" column shall denote either possible values
337 for the property, or requirements on the value formulation.

338 **Table 4 – DCIM_LCService - Properties**

Property Name	Type	Requirement	Additional Requirement
CreationClassName	String	Mandatory	The property value shall be "DCIM_LCService"
Name	String	Mandatory	The property value shall be "DCIM:LCService"
ElementName	String	Mandatory	The property value shall be "LC Service"
SystemCreationClassName	String	Mandatory	The property value shall be "DCIM_ComputerSystem"
SystemName	String	Mandatory	The property value shall be "DCIM:ComputerSystem"

339 **7.2 DCIM_LCEnumeration**

340 This section describes the implementation for the DCIM_LCEnumeration class.

341 This class shall be instantiated in the Implementation Namespace.

342 **7.2.1 Resource URIs for WinRM®**

343 The class Resource URI shall be “http://schemas.dell.com/wbem/wscim/1/cim-
344 schema/2/DCIM_LCEnumeration?__cimnamespace=root/dcim”

345 The key property shall be the InstanceID.

346 The instance Resource URI for DCIM_LCEnumeration instance shall be:

347 “http://schemas.dell.com/wbem/wscim/1/cim-

348 schema/2/DCIM_LCEnumeration?__cimnamespace=root/dcim+InstanceID=<InstanceID>”

349 **7.2.2 Operations**

350 The following table lists the operations implemented on DCIM_LCEnumeration.

351 **Table 2 – DCIM_LCEnumeration - Operations**

Operation Name	Requirements	Required Input
Get	Mandatory	Instance URI
Enumerate	Mandatory	Class URI
DCIM_LCService.SetAttribute()	Mandatory	See section 8.1
DCIM_LCService.SetAttributes()	Mandatory	See section 8.2

352

353 **7.2.3 Class Properties**

354 The following table lists the implemented properties for DCIM_LCEnumeration instance representing an
355 LC enumeration attribute. The “Requirements” column shall denote whether the property is implemented
356 (for requirement definitions, see section 3). The “Additional Requirement” column shall denote either
357 possible values for the property, or requirements on the value formulation.

Table 3 – Class: DCIM_LCEnumeration

Properties	Type	Requirement	Additional Requirements
InstanceID	string	Mandatory	The property value shall have a unique value.
AttributeName	String	Mandatory	The property value shall be from the “AttributeName” column in Table 4.
CurrentValue[]	string	Mandatory	The property value shall be one of the values in the “PossibleValues” column at the corresponding row in Table 4.
PendingValue[]	string	Mandatory	The property value shall be one of the values in the “PossibleValues” column at the corresponding row in Table 4.
DefaultValue[]	string	Mandatory	The property value shall represent the default value of the attribute and shall be one of the values in the “PossibleValues” column at the corresponding row in Table 4
IsReadOnly	boolean	Mandatory	The property value shall the value in the “IsReadOnly” column at the corresponding row in Table 4.
ElementName	string	Mandatory	The property value shall be the FQDD of the LC.
PossibleValues[]	string	Mandatory	The property value shall be equal to the array of the values in “PossibleValues” column at the corresponding row in Table 4.

359 The following table describes the requirements for the AttributeName, and PossibleValues properties for
 360 the DCIM_LCEnumeration class.
 361

Table 4 – DCIM_LCEnumeration Attributes

AttributeName	IsReadOnly	PossibleValues	Description
Licensed	True	<ul style="list-style-type: none"> • Yes • No 	Whether the Part Replacement feature is licensed.
Part Configuration Update	False	<ul style="list-style-type: none"> • Disabled • Apply always • Apply only if firmware match 	Part configuration update method.
Part Firmware Update	False	<ul style="list-style-type: none"> • Disable • Allow version upgrade only • Match firmware of replaced part 	<ul style="list-style-type: none"> • 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	False	<ul style="list-style-type: none"> • Disabled • Enabled 	<ul style="list-style-type: none"> • Enabled (default) – Enable collecting system inventory on restart • Disabled – Disable collecting inventory on restart
AutoDiscovery	True	<ul style="list-style-type: none"> • On • Off 	<ul style="list-style-type: none"> • Whether Auto Discovery feature is turned on or off.
Discovery Factory Defaults	True	<ul style="list-style-type: none"> • On • Off 	<ul style="list-style-type: none"> • Off (default) – Do not reset to factory defaults when performing auto discovery • On – Reset to factory defaults when performing auto discovery.
IPChangeNotifyPS	False	<ul style="list-style-type: none"> • On • Off 	<ul style="list-style-type: none"> • On – Notify provisioning server of an IP change. • Off – Do not notify provisioning server of an IP change.
VirtualAddressManagement	False	<ul style="list-style-type: none"> • Console • FlexAddress 	<ul style="list-style-type: none"> • Console – Console manages the virtual addresses • FlexAddress – (Blade only) Virtual addresses are managed through the flex addressing. <p>Note: If this attribute is configured through console, see to section “Setting virtual address attributes on CNA” in Simple NIC Profile.</p>
LifecycleControllerState	False	<ul style="list-style-type: none"> • Enabled • Disabled • Recovery 	<p>Enabled (Default)</p> <p>The LC core services shall be enabled to enable the update and inventory collection features.</p>

363 7.3 DCIM_LCString

364 This section describes the implementation for the DCIM_LCString class.

365 This class shall be instantiated in the Implementation Namespace.

366 **7.3.1 Resource URIs for WinRM®**

367 The class Resource URI shall be “http://schemas.dell.com/wbem/wscim/1/cim-
368 schema/2/DCIM_LCString?__cimnamespace=root/dcim”

369 The key property shall be the InstanceID.

370 The instance Resource URI for DCIM_LCString instance shall be:
371 “http://schemas.dell.com/wbem/wscim/1/cim-
372 schema/2/DCIM_LCString?__cimnamespace=root/dcim+InstanceID=<InstanceID>”

373 **7.3.2 Operations**

374 The following table details the implemented operations on DCIM_LCString.

375 **Table 5 – DCIM_LCString - Operations**

Operation Name	Requirements	Required Input
Get	Mandatory	Instance URI
Enumerate	Mandatory	Class URI
DCIM_LCService.SetAttributte()	Mandatory	See section 8.1
DCIM_LCService.SetAttributes()	Mandatory	See section 8.2

376 **7.3.3 Class Properties**

377 The following table lists the implemented properties for DCIM_LCString instance representing an LC
378 string attribute. The “Requirements” column shall denote whether the property is implemented (for
379 requirement definitions, see section 3). The “Additional Requirement” column shall denote either possible
380 values for the property, or requirements on the value formulation.

Table 6 – Class: DCIM_LCString

Properties	Type	Requirement	Additional Requirements
InstanceID	string	Mandatory	The property value shall have unique value.
AttributeName	string	Mandatory	The property value shall be from the "AttributeName" column in Table 7.
CurrentValue[]	string	Mandatory	The property shall represent the current value of the attribute.
PendingValue[]	string	Mandatory	The property value shall represent the pending value of the attribute.
DefaultValue[]	string	Mandatory	The property value shall represent the default value of the attribute.
IsReadOnly	boolean	Mandatory	The property value shall be the value in the "IsReadOnly" column at the corresponding row in Table 7.
ElementName	string	Mandatory	The property value shall be the FQDD of the LC.
MinLength	uint64	Mandatory	The property value shall be the value in the "MinLength" column at the corresponding row in Table 7. The CurrentValue and PendingValue shall have equal or higher number of characters than the MinLength property value from the corresponding row. The omission or NULL shall denote that no known constraint exists on the CurrentValue and PendingValue properties.
MaxLength	uint64	Mandatory	The property value shall be the value in the "MaxLength" column at the corresponding row in Table 7. The CurrentValue and PendingValue shall have equal or lower number of characters than the MaxLength property value. The omission or NULL shall denote that no known constraint exists on the CurrentValue and PendingValue properties.
StringType	uint32	Mandatory	The property shall describe the type of string in the CurrentValue, PendingValue, and DefaultValue properties and have one of the following values: <ul style="list-style-type: none"> • 0 - "Unknown", • 2 - "ASCII", • 3 - "hex", • 4 - "unicode", • 5 - "regex"

382 The following table describes possible DCIM_LCString attributes and the requirements for the
 383 AttributeName, MinLength, and MaxLength properties. The column headers represent the properties of
 384 the DCIM_LCString class. Each of the cells represents the values of the properties.

385 •

Table 7 – DCIM_LCString Attributes

AttributeName	IsRead Only	MinLength	MaxLength	Description
---------------	-------------	-----------	-----------	-------------

AttributeName	IsRead Only	MinLength	MaxLength	Description
SYSID	TRUE	N/A	N/A	Dell System ID.
Provisioning Server	FALSE	0	255	Provisioning server address and port. Attribute value examples: <ul style="list-style-type: none"> • Hostname • hostname.domain.com • 1.1.1.1 • Mypc:8080 • Mysps.dell.com(1.2.3.4):8080 • Host1:80,host2:8080, • Hostname[2001:db8:0000:1428:57ab]:443 [2001:db8:0000:1428:57ab]:443
VirtualAddressManagement Application*	FALSE*	0	32	The console name of the Virtual Address Management Application, if the DCIM_LCEnumeration VirtualAddressManagement attribute is set to CurrentValue equals "Console".

387 * - Note that VirtualAddressMangementApplication attribute is settable only if the
388 VirtualAddressManagement DCIM_LCEnumeration instance's CurrentValue is set to "Console".

389 7.4 Lifecycle Controller (LC) Management Registration – DCIM_LCRegistered 390 Profile

391 This section describes the implementation for the DCIM_LCRegisteredProfile class.

392 This class shall be instantiated in the Interop Namespace.

393 The DCIM_LCElementConformsToProfile association(s)' ConformantStandard property shall reference
394 the DCIM_LCRegisteredProfile instance.

395 7.4.1 Resource URIs for WinRM®

396 The class WBEM URI shall be "http://schemas.dmtf.org/wbem/wscim/1/cim-
397 schema/2/CIM_RegisteredProfile?__cimnamespace=root/interop"

398 The key property shall be the InstanceID property.

399 The instance WBEM URI shall be: "http://schemas.dell.com/wbem/wscim/1/cim-
400 schema/2/DCIM_LCRegisteredProfile?__cimnamespace=root/interop+InstanceID=
401 DCIM:LCManagement:1.1.0"

402 7.4.2 Operations

403 The following table lists the operations implemented on DCIM_LCRegisteredProfile.

404 **Table 8 – DCIM_LCRegisteredProfile - Operations**

Operation Name	Requirements	Required Input
----------------	--------------	----------------

Operation Name	Requirements	Required Input
Get	Mandatory	Instance URI
Enumerate	Mandatory	Class URI

405

406 7.4.3 Class Properties

407 The following table details the properties implemented for the DCIM_LCRegisteredProfile instance
 408 representing Lifecycle Controller (LC) Management implementation. The “Requirements” column shall
 409 denote whether the property is implemented (for requirement definitions, see section 3). The “Additional
 410 Requirements” column shall denote either possible values for the property, or requirements on the value
 411 formulation.

412

Table 9 – Class: DCIM_LCRegisteredProfile

Properties	Type	Requirement	Additional Requirements
InstanceID	string	Mandatory	This property shall have a value of “DCIM:LCManagement:1.1.0”
RegisteredName	string	Mandatory	This property shall have a value of “LC Management”.
RegisteredVersion	string	Mandatory	This property shall have a value of “1.6.0”.
RegisteredOrganization	uint16	Mandatory	This property shall have a value of 1 (Other).
OtherRegisteredOrganization	string	Mandatory	This property shall match “DCIM”
AdvertiseTypes[]	uint16	Mandatory	This property array shall have values “1(Other)” and “1(Other)”
AdvertiseTypeDescriptions[]	string	Mandatory	This property array shall have values “WS-Identify” and “Interop Namespace”
ProfileRequireLicense[]	string	Mandatory	This property array shall describe the required licenses for this profile. If no license is required for the profile, the property shall have value NULL.
ProfileRequireLicenseStatus[]	string	Mandatory	This property array shall contain the status for the corresponding license in the same element index of the ProfileRequireLicense array property. Each array element shall contain: <ul style="list-style-type: none"> • “LICENSED” • “NOT_LICENSED” If no license is required for the profile, the property shall have value NULL.

413

414 8 Methods

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

417 **8.1 DCIM_LCService.SetAttribute()**

418 The SetAttribute() method is used to set or change the value of an LC attribute.

419 Invoking the SetAttribute() method shall change the value of the DCIM_LCAAttribute.CurrentValue or
 420 DCIM_LCAAttribute.PendingValue property to the value specified by the AttributeValue parameter if the
 421 DCIM_LCAAttribute.IsReadOnly property is FALSE. Invoking this method when the
 422 DCIM_LCAAttribute.IsReadOnly property is TRUE shall result in no change to the value of the
 423 DCIM_LCAAttribute.CurrentValue property. The results of changing this value is described with the
 424 SetResult parameter.

425 Return code values for the SetAttribute() method are specified in Table 10 and parameters are specified
 426 in Table 11.

427 **NOTE:** Invoking the SetAttribute() method multiple times can result in the earlier requests being
 428 overwritten or lost.

429 **Table 10 – DCIM_LCService.SetAttribute() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

430 **Table 11 – DCIM_LCService.SetAttribute() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	AttributeName	string	DCIM_LCAAttribute.AttributeName
IN, REQ	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: <ul style="list-style-type: none"> • “Set CurrentValue” means CurrentValue property is set. • “Set PendingValue” means PendingValue property is set.
OUT	RebootRequired	string	A value of <ul style="list-style-type: none"> • “Yes” means a reboot is required to set this value. • “No” means a reboot is not required to set this value.
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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.

431

Table 12 – DCIM_LCService.SetAttribute() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	NA
LC003	Failed to change the FW Update mode	NA
LC004	PS info not formatted correctly	NA
LC005	Invalid FW update mode	NA
LC007	Invalid parameter	NA
LC008	Unsupported method parameter value	NA
LC012	Lifecycle Controller is being used by another process	NA
LC015	Invalid Part Configuration Update	NA
LC016	Missing required parameter	<parameter>
LC036	System Services is disabled	NA
LC041	Virtual Address Mgmt file error	NA
LC042	Virtual Address Mgmt using FlexAddr	NA
LC048	Invalid value of IPChangeNotifyPS	NA
LC049	Invalid value of VirtualAddressManagement	NA
LC057	Invalid AttributeName.	NA
LC058	Invalid AttributeValue for AttributeName.	<parameter>
LC059	Cannot set ReadOnly AttributeName.	<parameter>
LC062	An instance of Export or Import System Configuration is already running.	

433

434 8.2 DCIM_LCService.SetAttributes()

435 The SetAttributes() method is used to set or change the values of a group of attributes.

436 Invoking the SetAttributes() method shall change the values of the DCIM_LCAtribute.CurrentValue or
 437 PendingValue properties that correspond to the names specified by the AttributeName parameter and the
 438 values specified by the AttributeValue parameter if the respective DCIM_LCAtribute.IsReadOnly property
 439 is FALSE. Invoking the method when the respective DCIM_LCAtribute.IsReadOnly property is TRUE
 440 shall result in no change to the corresponding value of the DCIM_LCAtribute.CurrentValue property.

441 Return code values for the SetAttributes() method are specified in Table 13, and parameters are
 442 specified in Table 14.

443 **NOTE:** Invoking the SetAttributes() method multiple times can result in the earlier requests being
 444 overwritten or lost.

445

Table 13 – DCIM_LCService.SetAttributes() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred

446

Table 14 – DCIM_LCService.SetAttributes() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	AttributeName[]	string	The array parameter shall contain the AttributeName property values for the attributes to be modified.
IN, REQ	AttributeValue[]	string	<p>The array parameter shall contain the desired attribute values at the corresponding array element with the attribute name in the AttributeName array parameter.</p> <p>For example, if there are two attributes in the AttributeName array: AttributeName1 and AttributeName2, the array index 1 shall contain the value for AttributeName1 and array index 2 shall contain the value for AttributeName2.</p> <p>Attributes with array values shall not be set using this method, rather the SetAttribute() method shall be used to set the array attributes one at a time..</p> <p>If the value is valid, the CurrentValue or PendingValue property of the specified attribute shall be modified.</p>
OUT	SetResult[]	string	<p>Invoking the SetAttributes may result in the CurrentValue or PendingValue property of each input element being set. AttributeValue array shall have a corresponding SetResult value in the SetResult array.</p> <p>A return value of:</p> <ul style="list-style-type: none"> • “Set CurrentValue” means CurrentValue property is set. • “Set PendingValue” means PendingValue property is set.
OUT	RebootRequired[]	string	<p>Each input element’s AttributeValue array shall have a corresponding RebootRequired value in the RebootRequired array. A value of:</p> <ul style="list-style-type: none"> • “Yes” means a reboot is required to set this value. • “No” means a reboot is not required to set this value
OUT	MessageID[]	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
OUT	Message[]	string	Error Message in English corresponding to MessageID is returned if the method fails to execute

Qualifiers	Name	Type	Description/Values
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages

447

Table 15 – DCIM_LCService.SetAttributes() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	NA
LC003	Failed to change the FW Update mode	NA
LC004	PS info not formatted correctly	NA
LC005	Invalid FW update mode	NA
LC007	Invalid parameter	NA
LC008	Unsupported method parameter value	NA
LC012	Lifecycle Controller is being used by another process	NA
LC015	Invalid Part Configuration Update	NA
LC016	Missing required parameter	<parameter>
LC036	System Services is disabled	NA
LC041	Virtual Address Mgmt file error	NA
LC042	Virtual Address Mgmt using FlexAddr	NA
LC048	Invalid value of IPChangeNotifyPS	NA
LC049	Invalid value of VirtualAddressManagement	NA
LC057	Invalid AttributeName.	NA
LC058	Invalid AttributeValue for AttributeName.	<parameter>
LC059	Cannot set ReadOnly AttributeName.	<parameter>
LC062	An instance of Export or Import System Configuration is already running.	

448

449 **8.3 DCIM_LCService.CreateConfigJob()**

450 The CreateConfigJob() method is used to apply the pending values created by the SetAttribute and
 451 SetAttributes methods. The successful execution of this method creates a job for application of pending
 452 attribute values.

453 Return code values for the CreateConfigJob() method are specified in Table 16, and parameters are
 454 specified in Table 17.

455 **NOTE:** Subsequent calls to CreateConfigJob after the first CreateConfigJob will result in error until the
 456 first job is completed.

457

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

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

458

Table 17 – DCIM_LCService.CreateConfigJob() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status ¹
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

459

460

Table 18 – DCIM_LCService.CreateConfigJob() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	
LC012	Lifecycle Controller is being used by another process	NA
LC013	There are no pending values to set	NA
LC045	An instance of CreateConfigJob is already running	NA
JCP017	Maximum job limit reached, cannot create new jobs	NA
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	NA
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA
LC062	An instance of Export or Import System Configuration is already running.	

461

462

8.4 DCIM_LCService.ReInitiateDHS()

463

The ReInitiateDHS() method is used to reinitiate the provisioning server discovery and handshake.

464

Table 19 – DCIM_LCService.RelInitiateDHS() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred

465

Table 20 – DCIM_LCService.RelInitiateDHS() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	ProvisioningServer	string	An optional parameter to specify provisioning server addresses and ports used for auto discovery. If omitted, the Lifecycle Controller shall get the value from DHCP or DNS.
IN, REQ	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, REQ	PerformAutoDiscovery	uint16	<ul style="list-style-type: none"> 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 A/C cycle. NOTE: If NextBoot has value 3, then upon successful execution, the Discovery Factory Defaults attribute shall be set to "On".
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

466

Table 21 – DCIM_LCService.RelInitiateDHS() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	NA
LC004	PS info not formatted correctly	NA
LC008	Unsupported method parameter value	NA
LC009	Insufficient method parameters	NA

467

468 **8.5 DCIM_LCService.RelInitiateAutoDiscovery()**

469 The RelInitiateAutoDiscovery() method is used to reinitiate the discovery and handshake with static IP
470 configuration. If the method successfully executes, the iDRAC shall be in the static IP mode after the
471 auto-discovery is initiated and thus, shall not require DHCP setup. The static IP information for auto-
472 discovery can be set either through the optional input parameters of this method or through the iDRAC
473 attributes in the iDRAC Card Profile.

474 After the successful auto-discovery initiation, if the IPAddress parameter has non-NULL value, then the
475 iDRAC IPv4 address shall be set to that parameter's value, otherwise the iDRAC IPv4 Address shall be
476 set to the Address attribute in the IPv4 Information group in the iDRAC Card Profile.

477 After the successful auto-discovery initiation, if the Mask parameter has non-NULL value, then the iDRAC
478 IPv4 address net mask shall be set to that parameter's value, otherwise the iDRAC IPv4 address net
479 mask shall be set to the to the Netmask attribute in the IPv4 Information group in the iDRAC Card Profile.

480 After the successful auto-discovery initiation, if the Gateway parameter has non-NULL value, then the
481 iDRAC IPv4 gateway address shall be set to that parameter's value, otherwise the iDRAC IPv4 gateway
482 shall be set to the to the Gateway attribute in the IPv4 Information group in the iDRAC Card Profile.

483 After the successful auto-discovery initiation, if the DNS parameter has non-NULL value, then the iDRAC
484 IPv4 DNS address shall be set to that parameter's value, otherwise the iDRAC IPv4 DNS address shall
485 be set to the to the DNS1 attribute in the IPv4 Information group in the iDRAC Card Profile.

486 After the successful auto-discovery initiation, if the Domain parameter has non-NULL value, then the
487 iDRAC domain name shall be set to that parameter's value, otherwise the iDRAC domain name shall be
488 set to the to the DNSDomainName attribute in the NIC Information group in the iDRAC Card Profile.

Table 22 – DCIM_LCService.RelInitiateAutoDiscovery() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred

Table 23 – DCIM_LCService.RelInitiateAutoDiscovery() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	ProvisioningServer	string	An optional parameter to specify provisioning server addresses and ports used for auto discovery. If omitted, the Lifecycle Controller shall get the value from DNS.
IN, REQ	PerformAutoDiscovery	uint16	<ul style="list-style-type: none"> 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 A/C cycle. <p>NOTE: If NextBoot has value 3, then upon successful execution, the Discovery Factory Defaults attribute shall be set to "On".</p>
IN	IPAddress	string	<p>An optional parameter for specifying the iDRAC static IP configuration after the reset to the factory defaults occurs.</p> <p>The parameter shall be IPv4 address.</p> <p>The iDRAC IPv4 address shall be set to this parameter value after the reset occurs.</p> <p>NOTE: If this parameter is not specified then iDRAC shall be set to the value of the Address attribute in the IPv4 Information group in the iDRAC Card Profile.</p>
IN	Mask	string	<p>An optional parameter for specifying the iDRAC static IP configuration after the reset to the factory defaults occurs.</p> <p>The iDRAC shall be set to this net mask after the reset of iDRAC configuration.</p> <p>NOTE: If this parameter is not specified then iDRAC shall be set to the value of the Netmask attribute in the IPv4 Information group in the iDRAC Card Profile.</p>

Qualifiers	Name	Type	Description/Values
IN	Gateway	string	<p>An optional parameter for specifying the iDRAC static IP configuration after the reset to the factory defaults occurs.</p> <p>The parameter shall be IPv4 address.</p> <p>The iDRAC shall be set to this gateway IP address after the reset of iDRAC configuration.</p> <p>NOTE: If this parameter is not specified then iDRAC shall be set to the value of the Gateway attribute in the IPv4 Information group in the iDRAC Card Profile.</p>
IN	DNS	string	<p>An optional parameter for specifying the iDRAC static IP configuration after the reset to the factory defaults occurs.</p> <p>The parameter shall be IPv4 address.</p> <p>The iDRAC shall be set to this DNS IP address after the reset of iDRAC configuration.</p> <p>NOTE: If this parameter is not specified then iDRAC shall be set to the value of the DNS1 attribute in the IPv4 Information group in the iDRAC Card Profile.</p>
IN	Domain	string	<p>An optional parameter for specifying the iDRAC static IP configuration after the reset to the factory defaults occurs.</p> <p>The iDRAC shall be set to this domain after the reset of iDRAC configuration.</p> <p>NOTE: If this parameter is not specified then iDRAC shall be set to the value of the DNSDomainName attribute in the NIC Information group in the iDRAC Card Profile.</p>
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

491

Table 24 – DCIM_LCService.RelinitiateAutoDiscovery() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	NA
LC004	PS info not formatted correctly	NA

MessageID (OUT parameter)	Message	MessageArguments[]
LC008	Unsupported method parameter value	NA
LC009	Insufficient method parameters	NA
LC065	Static IP information is not configured correctly <attribute key>.	attribute key

492 **8.6 DCIM_LCService.ClearProvisioningServer()**

493 The ClearProvisioningServer() method is used to clear the provisioning server values.

494 **Table 25 – DCIM_LCService.ClearProvisioningServer() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

495 **Table 26 – DCIM_LCService.ClearProvisioningServer() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	MessageID	string	Error MessageID is returned if the method fails to execute
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

496 **Table 27 – DCIM_LCService.ClearProvisioningServer() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	

497 **8.7 DCIM_LCService.DownloadServerPublicKey()**

498 The DownloadServerPublicKey() method is used to set a public certificate in the chain of trust for
499 provisioning server. The successful invocation of this method shall set the certificate inside the
500 KeyContent parameter to the iDRAC trusted Root CA list. The certificate will be used to authenticate the
501 provisioning server access to iDRAC.

502

Table 28 – DCIM_LCService.DownloadServerPublicKey() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started

503

Table 29 – DCIM_LCService.DownloadServerPublicKey() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, OctetString, REQ	KeyContent	string	Base64 encoded public CA certificate.
OUT	Job	CIM_ConcreteJob REF	Returned to track the setting of the public key.
OUT	MessageID	string	Error Message ID can be used to index into Dell Message registry files.
OUT	Message	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages.

504

505

Table 30 – DCIM_LCService.DownloadServerPublicKey() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	NA
LC016	Missing required parameter	<parameter>
LC017	Invalid value of parameter	<parameter>
LC040	Resource Allocation Failure	
LC046	An instance of DownloadServerPublicKey is already running	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA

506

507 **8.8 DCIM_LCService.DownloadClientCerts()**

508 DCIM_LCService.DownloadClientCerts() is used to copy a CA cert and private key to the iDRAC to sign
509 the iDRAC certificate. When this command is issued the iDRAC will generate a new auto-discovery client

510 certificate and sign it using the CA cert and private key. The CA cert then needs to be added to the
511 provisioning server trusted Root CA list before performing auto-discovery.

512

Table 31 – DCIM_LCService.DownloadClientCerts() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started

513

Table 32 – DCIM_LCService.DownloadClientCerts() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, OctetString, REQ	KeyContent	string	Base64 encoded private key content.
IN, REQ	Password	string	Private key password.
IN, OctetString, REQ	CACContent	string	Base64 encoded root certificate content.
OUT	Job	CIM_ConcreteJob REF	Returned to track the setting of the public key.
OUT	MessageID	string	Error Message ID can be used to index into Dell Message registry files.
OUT	Message	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages.

514

Table 33 – DCIM_LCService.DownloadClientCerts() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	NA
LC016	Missing required parameter	<parameter>
LC017	Invalid value of parameter	<parameter>
LC040	Resource Allocation Failure	
LC047	An instance of DownloadClientCerts is already running	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA

515

516

8.9 DCIM_LCService.DeleteAutoDiscoveryClientCerts ()

517 The DeleteAutoDiscoveryClientCerts() method is used to wipe all configurations from the Lifecycle
 518 controller before the system is retired.

519 **Table 34 – DCIM_LCService.DeleteAutoDiscoveryClientCerts() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

520 **Table 35 – DCIM_LCService.DeleteAutoDiscoveryClientCerts() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

521 **Table 36 – DCIM_LCService.DeleteAutoDiscoveryClientCerts() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	

522

523 **8.10 DCIM_LCService.SetCertificateAndPrivateKey()**

524 The SetCertificateAndPrivateKey() method is used to set iDRAC owned server certificate and public and
 525 private key pairs using the contents of a PKCS#12 file. iDRAC will serve the certificate to identify itself
 526 during web server based connections such as for iDRAC's web application and WSMAN instrumentation
 527 connections. The certificate will also be served for connections with directory services.

528 **Table 37 – DCIM_LCService.SetCertificateAndPrivateKey() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

529 **Table 38 – DCIM_LCService.SetCertificateAndPrivateKey() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	Type	string	This parameter specifies the certificate use. Note: The parameter shall have "server" value.
IN, REQ	PKCS12	string	An input parameter that represents the Base64 encoded contents of PKCS#12 file to set. Note this is the content of the file and not a file name.
IN, REQ	PKCS12pin	string	Password to decode the PKCS12.
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.

530 **Table 39 – DCIM_LCService.SetCertificateAndPrivateKey() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	
LC016	Missing required parameter	"Type", "PKCS12", "PKCS12pin"
LC017	Invalid value of parameter	"Type", "PKCS12", "PKCS12pin"

531
532

533 **8.11 DCIM_LCService.SetPublicCertificate()**

534 The SetPublicCertificate() method is used to set a public certificate in the chain of trust for directory
 535 services. The successful invocation of this method shall set the certificate inside the Certificate parameter
 536 to the iDRAC trusted Root CA list. The certificate will be used to authenticate the directory service access
 537 to iDRAC.

538 **Table 40 – DCIM_LCService.SetPublicCertificate() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

539 **Table 41 – DCIM_LCService.SetPublicCertificate() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	Type	string	This parameter specifies the service the certificate is for. "directoryCA" = CA certificate for Active Directory or LDAP server
IN, REQ	Certificate	string	An input parameter that represents the certificate to set. The certificate must be in X509 format and Base64 encoded.
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

540 **Table 42 – DCIM_LCService.SetPublicCertificate() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	
LC016	Missing required parameter %s	"Type", "Certificate"
LC017	Invalid value of parameter %s	"Type", "Certificate"

541 **8.12 DCIM_LCService.DeleteAutoDiscoveryServerPublicKey()**

542 The DeleteAutoDiscoveryServerPublicKey() method is used to delete the server public key set previously
 543 by the auto discovery method.

544 **Table 43 – DCIM_LCService.DeleteAutoDiscoveryServerPublicKey() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

545 **Table 44 – DCIM_LCService.DeleteAutoDiscoveryServerPublicKey() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

546 **Table 45 – DCIM_LCService.DeleteAutoDiscoveryServerPublicKey() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	

547

548 **8.13 DCIM_LCService.InsertCommentInLCLog()**

549 The InsertCommentInLCLog() method is used to insert additional user comments into the Lifecycle
550 Controller log.

551 **Table 46 – DCIM_LCService.InsertCommentInLCLog() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

552 **Table 47 – DCIM_LCService.InsertCommentInLCLog() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	Comment	string	This parameter holds the text that shall be inserted into the LC log Comment field.
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

553

554

555 **Table 48 – DCIM_LCService.InsertCommentInLCLog() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	
LC008	Unsupported method parameter value	
LC014	Comment length must be between 1 and 255 characters	

556 **8.14 DCIM_LCService.ExportLCLog()**

557 The ExportLCLog() method is used to export the log from the Lifecycle Controller to a remote share.

558 **Table 49 – DCIM_LCService.ExportLCLog() Method: Return Code Values**

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

559 **Table 50 – DCIM_LCService.ExportLCLog() Method: Parameters**

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	string	The IP address of the target export server.
IN, REQ	ShareName	string	The directory path to the mount point.
IN, REQ	FileName	string	The target output file name.
IN, REQ	ShareType	uint16	Type of share: NFS=0, CIFS=2
IN	Username	String	User name for the target export server.
IN	Password	String	Password for the target export server.
IN	Workgroup	String	The applicable workgroup.
OUT	Job	CIM_ConcreteJob REF	[1] Returned to keep track of config job status ¹ .
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.

560

561

562

Table 51 – DCIM_LCService.ExportLCLog() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter <parameter name>	Parameter name, see Table 52
LC017	Invalid value of parameter <parameter name>	Parameter name, see Table 53
LC022	LCL Export was successful	
LC023	Cannot access network share	
LC030	Filepath is a read-only file system	
LC037	An instance of LCExport is already running	
LC040	Resource Allocation Failure	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA

563

564 **8.15 DCIM_LCService.ExportHWInventory()**

565 The ExportHWInventory() method is used to export the hardware inventory from the Lifecycle Controller
566 to a remote share.

567

Table 54 – DCIM_LCService.ExportHWInventory() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

568

Table 55 – DCIM_LCService.ExportHWInventory() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	string	The IP address of the target export server.
IN, REQ	ShareName	string	The directory path to the mount point.
IN, REQ	FileName	string	The target output file name.
IN, REQ	ShareType	uint16	Type of share: NFS=0, CIFS=2
IN	XMLSchema	uint8	The schema to be used for the Hardware Inventory XML output: 0 (CIM-XML), 1(Simple). If this parameter is not specified, then the output shall be in CIM-XML schema XML.
IN	Username	String	User name for the target export server.
IN	Password	String	Password for the target export server.
IN	Workgroup	String	The applicable workgroup.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status ¹
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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.

569

570

571

Table 56 – DCIM_LCService.ExportHWInventory() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter <parameter name>	Parameter name, see Table 57
LC017	Invalid value of parameter <parameter name>	Parameter name, see Table 58
LC023	Cannot access network share	
LC025	Insufficient space on network share	
LC027	Hardware Inventory Export was	

MessageID (OUT parameter)	Message	MessageArguments[]
	successful	
LC029	Filepath is invalid	
LC030	Filepath is a read-only file system	
LC038	An instance of Hardware Inventory is already running	
LC040	Resource Allocation Failure	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA

572

573 **8.16 DCIM_LCService.ExportFactoryConfiguration()**

574 The ExportFactoryConfiguration() method is used to export the factory configuration from the Lifecycle
575 Controller to a remote share.

576

Table 59 – DCIM_LCService.ExportFactoryConfiguration() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

577

Table 60 – DCIM_LCService.ExportFactoryConfiguration() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	string	The IP address of the target export server.
IN, REQ	ShareName	string	The directory path to the mount point.
IN, REQ	FileName	string	The target output file name.
IN, REQ	ShareType	uint16	Type of share: NFS=0, CIFS=2
IN	Username	String	User name for the target export server.
IN	Password	String	Password for the target export server.
IN	Workgroup	String	The applicable workgroup.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status.
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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.

578

Table 61 – DCIM_LCService.ExportFactoryConfiguration() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter <parameter name>	Parameter name, see Table 62
LC017	Invalid value of parameter <parameter name>	Parameter name, see Table 63
LC023	Cannot access network share	
LC025	Insufficient space on network share	
LC029	Filepath is invalid	
LC030	Filepath is a read-only file system	
LC033	Factory Configuration Export was successful	
LC034	Factory Configuration was not found on the system	
LC039	An instance of Factory	

MessageID (OUT parameter)	Message	MessageArguments[]
	Configuration is already running	
LC040	Resource Allocation Failure	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA

579

580 **8.17 DCIM_LCService.ExportSystemConfiguration()**

581 This method is used to export the system configuration from the Lifecycle Controller to a file on a remote
582 share.

583

Table 64 – DCIM_LCService.ExportSystemConfiguration() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

584

Table 65 – DCIM_LCService.ExportSystemConfiguration() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	string	The IP address of the target export server.
IN, REQ	ShareName	string	The directory path to the mount point.
IN, REQ	FileName	string	The target output file name.
IN, REQ	ShareType	uint16	Type of share: NFS=0, CIFS=2
IN	Username	String	User name for the target export server.
IN	Password	String	Password for the target export server.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status.
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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.

585

Table 66 – DCIM_LCService.ExportSystemConfiguration() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter <parameter name>	Parameter name, see Table 65
LC017	Invalid value of parameter <parameter name>	Parameter name, see Table 65
LC023	Cannot access network share	
LC040	Resource Allocation Failure	
LC062	An instance of Export or Import System Configuration is already running.	
LC063	Cannot create new jobs until the existing running jobs are completed or deleted.	
LC064	Parameter exceeds maximum length <max parameter length> for <parameter name>	Parameter name, see Table 65
JCP017	Maximum job limit reached, cannot create new jobs	

586

587 8.18 DCIM_LCService.ImportSystemConfiguration()

588 This method is used to import the system configuration from the Lifecycle Controller from a file on a
 589 remote share. The successful invocation of this method shall return a job. The job execution results shall
 590 be logged in the Lifecycle log as part of configuration results. The Dell Record Log Profile provides the
 591 DCIM_LCRecordLog.GetConfigResults() method to retrieve the configuration results from the lifecycle
 592 logs based on the returned job's JobID (see Dell Record Log Profile for more details). Table 67 –
 593

DCIM_LCService.ImportSystemConfiguration() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

594

Table 68 – DCIM_LCService.ImportSystemConfiguration() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	string	The IP address of the target import server.
IN, REQ	ShareName	string	The directory path to the mount point.
IN, REQ	FileName	string	The target output file name.
IN, REQ	ShareType	uint16	Type of share: NFS=0, CIFS=2
IN	ShutdownType	uint16	Type of the host shut down before perform the import operation. Type of shut down: Graceful=0, Forced =1
IN	TimeToWait	uint16	The time to wait for the host to shut down. Default and minimum value is 300 seconds. Maximum value is 3600 seconds.
IN	EndHostPowerState	uint16	The desired host power state after the import operation is finished. Default value is 1 - On. Type of state: Off=0, On=1
IN	Username	String	User name for the target import server.
IN	Password	String	Password for the target import server.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status.
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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.

595

Table 69 – DCIM_LCService.ImportSystemConfiguration() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter <parameter name>	Parameter name, see Table 65

MessageID (OUT parameter)	Message	MessageArguments[]
LC017	Invalid value of parameter <parameter name>	Parameter name, see Table 65
LC023	Cannot access network share	
LC040	Resource Allocation Failure	
LC062	An instance of Export or Import System Configuration is already running.	
LC063	Cannot create new jobs until the existing running jobs are completed or deleted.	
LC064	Parameter exceeds maximum length <max parameter length> for <parameter name>	Parameter name, see Table 65
JCP017	Maximum job limit reached, cannot create new jobs	

596

597

598 **8.19 DCIM_LCService.LCWipe()**

599 This LCWipe() method is used to delete all configurations from the Lifecycle controller before the system
600 is retired.

601

602

Table 70 – DCIM_LCService.LCWipe() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred

603

Table 71 – DCIM_LCService.LCWipe() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

604

Table 72 – – DCIM_LCService.LCWipe() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC012	Lifecycle Controller is being used by another process	
LC044	An instance of LCWipe is already running	

605

8.20 DCIM_LCService.TestNetworkShare()

606

The TestNetworkShare() method is used to test network connection for the Lifecycle Controller.

607

Table 73 – DCIM_LCService.TestNetworkShare() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred

Table 74 – DCIM_LCService.TestNetworkShare() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	IPAddress	String	IP address of the NFS or CIFS share such as 192.168.10.12. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	ShareName	String	Share name for the network share such as "imageshare". The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	ShareType	Unit16	Type of the share or storage. <ul style="list-style-type: none"> • 0 – NFS • 2 – CIFS If ShareType is blank or null, it shall be considered as NFS type. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	Username	String	User name for the remote share
IN	Password	String	Password for the remote share
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

Table 75 – DCIM_LCService.TestNetworkShare() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter	<parameter>
LC017	Invalid value of parameter	"ShareType"
LC023	Cannot access network share	NA

8.21 DCIM_LCService.BackupImage()

The BackupImage() method is used to backup firmware and configurations for the Lifecycle Controller.

Table 76 – DCIM_LCService.BackupImage() Method: Return Code Values

Value	Description
2	Error occurred
4096 ¹	A DCIM_ConcreteJob is returned ¹

Table 77 – DCIM_LCService.BackupImage() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	IPAddress	String	IP address of the NFS or CIFS share such as 192.168.10.12. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	ShareName	String	Share name for the network share such as "imageshare". The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	ShareType	Unit16	Type of the share or storage. <ul style="list-style-type: none"> • 0 – NFS • 2 – CIFS • 4 – VFLASH If ShareType is blank or null, it shall be considered as NFS type. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	Passphrase	String	The passphrase for the image. The parameter shall be required, if the ShareType parameter has value 0 (NFS), or 2 (CIFS) or not specified.
IN	ImageName	String	The name of the backup file. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	Username	String	User name for the remote share
IN	Password	String	Password for the remote share
IN	Workgroup	String	Workgroup for the share
IN	ScheduledStartTime	String	Start time for the job execution in format: yyymmddhhmmss. The string "TIME_NOW" means immediate.
IN	UntilTime	String	End time for the job execution in format: yyymmddhhmmss. : If this parameter is not NULL, then ScheduledStartTime parameter shall also be specified.
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
OUT	Job	DCIM_ConcreteJob Ref	The job that represents this operation ¹
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

614 NOTE: 1 – If return code is 4096 (Job Created), the newly created job will not execute if the LC core services are not
 615 running (DCIM_LCEnumeration with AttributeName equal to “LifecycleControllerState” has the CurrentValue property
 616 equal to “Disabled”).

617 **Table 78 – DCIM_LCService.BackupImage() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	
LC017	Invalid value of parameter	"ShareType"
LC023	Cannot access network share	NA
LC051	Cannot create multiple Export jobs.	NA
SUP017	Invalid start time.	
SUP018	Invalid until time.	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	

618

619 **8.22 DCIM_LCService.RestoreImage()**

620 This RestoreImage() method is used to restore firmware and configurations.

621

Table 79 – DCIM_LCService.RestoreImage() Method: Return Code Values

Value	Description
2	Error occurred
4096 ¹	A DCIM_ConcreteJob is returned ¹

622

Table 80 – DCIM_LCService.RestoreImage() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	IPAddress	String	IP address of the NFS or CIFS share. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN.	ShareName	String	Share name for the network share, The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	ShareType	UInt16	Type of the share or storage. <ul style="list-style-type: none"> • 0 – NFS • 2 – CIFS • 4 – VFLASH If ShareType is blank or null, it shall be considered as NFS type
IN	Passphrase	String	The passphrase for the image. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	ImageName	String	The name of the backup file. The parameter shall be required, if the ShareType parameter has value 0 (NFS), 2 (CIFS), or not specified.
IN	Username	String	User name for the remote share
IN	Password	String	Password for the remote share
IN	Workgroup	String	Workgroup for the share
IN, REQ	PreserveVDConfig	UInt16	Whether to preserve the VD (Virtual Disk) configuration.
IN	ScheduledStartTime	String	Start time for the job execution in format: yyyyymmddhhmmss. The string "TIME_NOW" means immediate.
IN	UntilTime	String	End time for the job execution in format: yyyyymmddhhmmss. : If this parameter is not NULL, then ScheduledStartTime parameter shall also be specified.
OUT	Job	DCIM_ConcreteJob Ref	The job that represents this operation ¹
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.

Qualifiers	Name	Type	Description/Values
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

623 NOTE: 1 – If return code is 4096 (Job Created), the newly created job will not execute if the LC core services are not
624 running (DCIM_LCEnumeration with AttributeName equal to “LifecycleControllerState” has the CurrentValue property
625 equal to “Disabled”).

626 **Table 81 – DCIM_LCService.RestoreImage() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC002	General failure	LC002
LC017	Invalid value of parameter	“ShareType”, “PreserveVDCConfig”
LC023	Cannot access network share	NA
LC052	Cannot create multiple Import jobs.	NA
SUP017	Invalid start time	
SUP018	Invalid until time	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	

627

628 **8.23 DCIM_LCService.GetRSStatus()**

629 The GetRSStatus() method is used to obtain the Data Manager (Remote Services) status.

630 **NOTE:** The GetRemoteServicesAPIStatus() method execution reports more granular and detailed status
631 of the Remote Services API.

632

Table 82 – DCIM_LCService.GetRSStatus() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred

633

Table 83 – DCIM_LCService.GetRSStatus() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	Status	string	The status for the Data Manager: Ready Not Ready Reloading
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

634

Table 84 – DCIM_LCService.GetRSStatus() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
RSI0001	The remote service is available	
RSI0002	The remote service is currently not available	
RSI0003	The remote service is reloading data	
RSI0004	The remote service is unavailable	

635

8.24 DCIM_LCService.GetRemoteServicesAPIStatus ()

636 The GetRemoteServicesAPIStatus() method is used to obtain the overall remote services API status that
637 includes both the host system status as well as the remote services (Data Manager) status. The overall
638 rolled up status shall be reflected in the Status output parameter.

639 **NOTE:** The LCStatus output parameter value includes the status reported by the DMStatus output
640 parameter in the GetRSStatus() method and the Lifecycle Controller status. Thus, GetRSStatus() method
641 invocation is redundant.

642 **Table 67 – DCIM_LCService.GetRemoteServicesAPIStatus() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred

643 **Table 68 – DCIM_LCService.GetRemoteServicesAPIStatus() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	ServerStatus	uint8	The host system status: <ul style="list-style-type: none"> • 0 – Powered Off • 1 – In POST • 2 – Out of POST • 3 – Collecting System Inventory • 4 – Automated Task Execution • 5 – Lifecycle Controller Unified Server Configurator
OUT	LCStatus	uint8	The Lifecycle Controller status that includes the Data Manager status: <ul style="list-style-type: none"> • 0 – Ready • 1 – Not Initialized • 2 – Reloading data • 3 – Disabled • 4 – In Recovery • 5 – In Use
OUT	Status	uint8	The overall status of the Remote Services API: <ul style="list-style-type: none"> • 0 – Ready • 1 – Not Ready
OUT	MessageID	string	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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

644 **Table 85 – DCIM_LCService.GetRemoteServicesAPIStatus() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
LC060	Lifecycle Controller Remote Services is not ready.	NA
LC061	Lifecycle Controller Remote Services is ready.	NA

645 **8.25 DCIM_LCService.ExportCertificate()**

646 The ExportCertificate() method is used to export unique generated SSL Certificates from the Lifecycle
647 Controller to a file in remote share.

648

Table 71 – DCIM_LCService.ExportCertificate() Method: Return Code Values

Value	Description
2	Error occurred
4096	Job started: REF returned to started CIM_ConcreteJob

649

Table 72 – DCIM_LCService.ExportCertificate() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Type	uint16	The type of the certificate 2 = Custom Signing Certificate
IN, REQ	IPAddress	string	The IP address of the target export server.
IN, REQ	ShareName	string	The directory path to the mount point.
IN, REQ	ShareType	uint16	Type of share: NFS=0, CIFS=2
IN	Username	String	User name for the target export server.
IN	Password	String	Password for the target export server.
IN	Workgroup	String	The applicable workgroup.
OUT	Job	CIM_ConcreteJob REF	Returned to keep track of config job status.
OUT	MessageID	String	Error Message ID may be used to look-up in the Dell Message registry files. For more information, see Error Message Registry.
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.

650

Table 73 – DCIM_LCService.ExportCertificate() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
LC016	Missing required parameter <parameter name>	Parameter name, see Table 72
LC017	Invalid value of parameter <parameter name>	Parameter name, see Table 72
LC023	Cannot access network share	
LC025	Insufficient space on network share	
LC029	Filepath is invalid	
LC030	Filepath is a read-only file system	
LC040	Resource Allocation Failure	
LC064	The value exceeds the maximum length of %d characters for %s	
LC065	Static IP information is not	

MessageID (OUT parameter)	Message	MessageArguments[]
	configured correctly	
LC066	An instance of Export Certificate is already running	
LC067	Export SSL Certificate success	
LC069	Certificate does not Exist on the System.	
JCP017	Maximum job limit reached, cannot create new jobs	
JCP018	Backup Job is running, cannot create new jobs until the existing Backup job is completed or is cancelled	
JCP019	Restore Job is running, cannot create new jobs until the existing Restore job is completed or is cancelled	NA

651

652 **9 Use Cases**

653 See *Lifecycle Controller (LC) Integration Best Practices Guide*

654 **10 CIM Elements**

655 No additional requirements have been defined.

656 **11 Privilege and License Requirement**

657 The following table describes the privilege and license requirements for the listed operations . For the
658 detailed explanation of the privileges and licenses, refer to the Dell WSMAN Licenses and Privileges
659 specification.

660

Table 74 – Privilege and License Requirements

Class and Method	Instance (GroupID.Attribute Name)	Operation	User Privilege Required	License Required
DCIM_LCString	SYSID	ENUMERATE, GET	Login	NONE
DCIM_LCEnumeration	Licensed	ENUMERATE, GET	Login	NONE
DCIM_LCString	CMCIP			NONE

DCIM_LCString	Provisioning Server	ENUMERATE, GET	Login	LM_AUTO_DISCOVERY
DCIM_LCEnumeration	Auto Discovery	ENUMERATE, GET	Login	LM_AUTO_DISCOVERY
DCIM_LCEnumeration	Discovery Factory Defaults	ENUMERATE, GET	Login	LM_AUTO_DISCOVERY
DCIM_LCEnumeration	Part Configuration Update	ENUMERATE, GET	Login	LM_PART_REPLACEMENT
DCIM_LCEnumeration	Part Firmware Update	ENUMERATE, GET	Login	LM_PART_REPLACEMENT
DCIM_LCEnumeration	Collect System Inventory on Restart	ENUMERATE, GET	Login	LM_REMOTE_CONFIGURATION
DCIM_LCEnumeration	VirtualAddressManagement	ENUMERATE, GET	Login	LM_REMOTE_CONFIGURATION
DCIM_LCString	VirtualAddressManagementApplication	ENUMERATE, GET	Login	LM_REMOTE_CONFIGURATION
DCIM_LCEnumeration	IPChangeNotifyPS	ENUMERATE, GET	Login	LM_AUTO_DISCOVERY
DCIM_LCEnumeration	Lifecycle Controller State	ENUMERATE, GET	Login	LM_REMOTE_CONFIGURATION
DCIM_LCService		ENUMERATE, GET	Login	NONE
DCIM_LCService. SetAttribute()		INVOKE	Login, System Control	LM_REMOTE_CONFIGURATION
DCIM_LCService. SetAttributes()		INVOKE	Login, System Control	LM_REMOTE_CONFIGURATION
DCIM_LCService. CreateConfigJob()		INVOKE	Login, System Control	LM_REMOTE_CONFIGURATION
DCIM_LCService. ReInitiateDHS()		INVOKE	Login, System Control	LM_AUTO_DISCOVERY
DCIM_LCService. ReInitiateAutoDiscovery()		INVOKE	Login, System Control	LM_AUTO_DISCOVERY
DCIM_LCService. ClearProvisioningServer()		INVOKE	Login, System Control	LM_AUTO_DISCOVERY
DCIM_LCService. DownloadServerPublicKey()		INVOKE	Login, System Control	LM_REMOTE_CONFIGURATION

DCIM_LCService. DownloadClientCerts()		INVOKE	Login, System Control	LM_REMOTE_CONFI GURATION
DCIM_LCService. DeleteAutoDiscoveryClientCert s()		INVOKE	Login, System Control	LM_AUTO_DISCOVE RY
DCIM_LCService. SetCertificateAndPrivateKey()		INVOKE	Login, System Control	LM_REMOTE_CONFI GURATION
DCIM_LCService. SetPublicCertificate()		INVOKE	Login, System Control	LM_REMOTE_CONFI GURATION
DCIM_LCService. DeleteAutoDiscoveryServerPu blicKey()		INVOKE	Login, System Control	LM_AUTO_DISCOVE RY
DCIM_LCService.LCWipe()		INVOKE	Login, System Control	NONE
DCIM_LCService.LCWipe().Ex portFactoryConfiguration()		INVOKE	Login	LM_BASIC_REMOTE _INVENTORY_EXPO RT
DCIM_LCService. ExportHWInventory()		INVOKE	Login	LM_BASIC_REMOTE _INVENTORY_EXPO RT
DCIM_LCService. ExportLCLog()		INVOKE	Login	NONE
DCIM_LCService.. InsertCommentInLCLog()		INVOKE	Login, System Operations	LM_REMOTE_CONFI GURATION
DCIM_LCService. GetRSStatus()		INVOKE	Login	NONE
DCIM_LCService. GetRemoteServicesAPIStatus ()		INVOKE	Login	NONE
DCIM_LCService.TestNetwork Share()		INVOKE	Login	LM_REMOTE_CONFI GURATION
DCIM_LCService.BackupImag e()		INVOKE	Login, System Control	LM_BACKUP_REST ORE
DCIM_LCService.RestoreImag e()		INVOKE	Login, System Control	LM_RESTORE
DCIM_LCService.ExportCertifi cate()		INVOKE	Login	LM_REMOTE_CONFI GURATION

DCIM_LCService.ExportSystemConfiguration()		INVOKE	Login, System Control	LM_REMOTE_CONFIGURATION
DCIM_LCService.ImportSystemConfiguration()		INVOKE	Administrator	LM_REMOTE_CONFIGURATION
DCIM_LCRegisteredProfile		ENUMERATE, GET	Login	None.
DCIM_LCElementConformsToProfile		ENUMERATE, GET	Login	None.

661
662
663
664
665

ANNEX A (informative)

Change Log

Version	Date	Description
1.6.0		Added the IPAddress, Mask, Gateway, DNS, Domain optional parameters to the DCIM_LCService.RelInitiateDHS() method. Added the DCIM_LCService.ExportSystemConfiguration() method.
1.6.0	4/23/2012	Added optional MaxExecutionTime input parameter to DCIM_LCService.ExportSystemConfiguration() method. Added messages LC016 and LC017 to methods - DCIM_LCService.ExportLCLog(), DCIM_LCService.ExportHWInventory(), DCIM_LCService.ExportFactoryConfiguration() and DCIM_LCService.ExportSystemConfiguration(). Added LC062 error message to the SetAttribute(), SetAttributes(), and CreateTargetedConfigJob() methods.
1.6.0	4/25/2012	Add licensing and privilege requirement for ExportSystemConfiguration().
1.6.0	4/30/2012	Remove MaxExecutionTime input parameter from DCIM_LCService.ExportSystemConfiguration() method.
1.6.0	5/08/2012	Added DCIM_LCService.RelInitiateAutoDiscovery() method to support Auto-Discovery with static IP. Removed optional parameters (IPAddress, Mask, Gateway, DNS, Domain) from DCIM_LCService.RelInitiateDHS().
1.6.0	6/05/2012	Added DCIM_LCService.ExportCertificate method to support export of uniquely generated SSL certificates
1.6.1	6/18/2012	Added DCIM_LCService.ImportSystemConfiguration() method to support importing system configuration settings.
1.6.2	6/25/2012	Added DCIM_LCService.TestNetworkShare() method to test network connection.
1.6.2	7/10/2012	Added Target to DCIM_LCService.ExportSystemConfiguration() method. Added ScheduledStartTime and UntilTime to DCIM_LCService.ImportSystemConfiguration() method.
1.6.2	7/25/2012	Changed privilege requirement to Administrator for DCIM_LCService.ImportSystemConfiguration() method.

666
667

668

669

670