

1 Operating System (OS)

2 Deployment Profile

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23 **Document Number: DCIM1035**

24 **Document Type: Specification**

25 **Document Status: Published**

26 **Document Language: E**

26 **Date: 2012-03-08**

27 **Version: 1.4.0**

28

29

30



31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66

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 – 2012 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

67			
68	1	Scope	7
69	2	Normative References.....	7
70	3	Terms and Definitions	8
71	4	Symbols and Abbreviated Terms	9
72	5	Synopsis.....	9
73	6	Description	10
74	7	Implementation Requirements	12
75	7.1	DCIM_OSDeploymentService	12
76	7.2	DCIM_OSConcreteJob	13
77	7.3	Operating System (OS) Deployment Profile Registration – DCIM_LCRegistered Profile.....	15
78	8	Methods.....	17
79	8.1	DCIM_OSDeploymentService.GetDriverPackInfo()	17
80	8.2	DCIM_OSDeploymentService.UnpackAndAttach()	18
81	8.3	DCIM_OSDeploymentService.DetachDrivers()	20
82	8.4	DCIM_OSDeploymentService.UnpackAndShare()	20
83	8.5	DCIM_OSDeploymentService.BootToNetworkISO().....	22
84	8.6	DCIM_OSDeploymentService.DetachISOImage()	24
85	8.7	DCIM_OSDeploymentService.BootToPXE()	25
86	8.8	Method: DCIM_OSDeploymentService.BootToHD().....	26
87	8.9	DCIM_OSDeploymentService.DownloadISOToVFlash()	27
88	8.10	DCIM_OSDeploymentService.BootToISOFromVFlash()	29
89	8.11	DCIM_OSDeploymentService.DetachISOFromVFlash()	30
90	8.12	DCIM_OSDeploymentService.DeleteISOFromVFlash()	31
91	8.13	DCIM_OSDeploymentService.ConnectNetworkISOImage().....	32
92	8.14	DCIM_OSDeploymentService.DisconnectNetworkISOImage()	34
93	8.15	DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo()	34
94	8.16	DCIM_OSDeploymentService.SkipISOImageBoot()	35
95	8.17	DCIM_OSDeploymentService.DisconnectRFSISOImage()	36
96	8.18	DCIM_OSDeploymentService.ConnectRFSISOImage()	37
97	8.19	DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo()	39
98	8.20	DCIM_OSDeploymentService.GetHostMACInfo()	39
99	9	Use Cases.....	40
100	10	CIM Elements.....	40
101	11	Privilege and License Requirement	40
102			

103	Figures	
104	Figure 1 – Operating System (OS) Deployment Profile: Class Diagram	11
105		

106	Tables	
107	Table 1 – Related Profiles	10
108	Table 2 – Class Requirements: Power State Management Profile.....	12
109	Table 3 –DCIM_OSDeploymentService - Operations	13
110	Table 3 – DCIM_OSDeploymentService - Properties.....	13
111	Table 4 – DCIM_OSConcreteJob - Properties	14
112	Table 5 – Job Descriptions.....	14
113	Table 6 – DCIM_LCRegisteredProfile - Operations.....	16
114	Table 7 – Class: DCIM_LCRegisteredProfile.....	16
115	Table 8 – DCIM_OSDeploymentService.GetDriverPackInfo() Method: Return Code Values.....	17
116	Table 9 – DCIM_OSDeploymentService.GetDriverPackInfo() Method: Parameters.....	18
117	Table 10 – DCIM_OSDeploymentService.GetDriverPackInfo() Method: Standard Messages	18
118	Table 11 – DCIM_OSDeploymentService.UnpackAndAttach() Method: Return Code Values	18
119	Table 12 – DCIM_OSDeploymentService.UnpackAndAttach() Method: Parameters	19
120	Table 13 – DCIM_OSDeploymentService.UnpackAndAttach() Method: Standard Messages.....	19
121	Table 14 – DCIM_OSDeploymentService.DetachDrivers() Method: Return Code Values	20
122	Table 15 – DCIM_OSDeploymentService.DetachDrivers() Method: Parameters	20
123	Table 16 – DCIM_ OSDeploymentService.DetachDrivers() Method: Standard Messages.....	20
124	Table 17 – DCIM_OSDeploymentService.UnpackAndShare() Method: Return Code Values.....	20
125	Table 18 – DCIM_OSDeploymentService.UnpackAndShare() Method: Parameters.....	21
126	Table 19 – DCIM_ OSDeploymentService.UnpackAndShare() Method: Standard Messages	21
127	Table 20 – DCIM_OSDeploymentService.BootToNetworkISO() Method: Return Code Values	22
128	Table 21 – DCIM_OSDeploymentService.BootToNetworkISO() Method: Parameters.....	23
129	Table 22 – DCIM__OSDeploymentService.BootToNetworkISO() Method: Standard Messages	23
130	Table 23 – DCIM_OSDeploymentService.DetachISOImage() Method: Return Code Values	24
131	Table 24 – DCIM_OSDeploymentService.DetachISOImage() Method: Parameters	25
132	Table 25 – DCIM_OSDeploymentService.DetachISOImage() Method: Standard Messages.....	25
133	Table 26 – DCIM_OSDeploymentService.BootToPXE() Method: Return Code Values	25
134	Table 27 – DCIM_OSDeploymentService.BootToPXE() Method: Parameters	26
135	Table 28 – DCIM_OSDeploymentService.BootToPXE() Method: Standard Messages.....	26
136	Table 29 – DCIM_OSDeploymentService.BootToHD() Method: Return Code Values	26
137	Table 30 – DCIM_OSDeploymentService.BootToHD() Method: Parameters	26
138	Table 31 – DCIM_OSDeploymentService.BootToHD() Method: Standard Messages.....	27
139	Table 32 – DCIM_OSDeploymentService.DownloadISOToVFlash() Method: Return Code Values	27
140	Table 33 – DCIM_OSDeploymentService.DownloadISOToVFlash() Method: Parameters	28
141	Table 34 – DCIM_OSDeploymentService.DownloadISOToVFlash() Method: Standard Messages.....	28
142	Table 35 – DCIM_OSDeploymentService.BootToISOFromVFlash() Method: Return Code Values.....	29
143	Table 36 – DCIM_OSDeploymentService.BootToISOFromVFlash() Method: Parameters	29
144	Table 37 – DCIM_OSDeploymentService.BootToISOFromVFlash() Method: Standard Messages	29
145	Table 38 – DCIM_OSDeploymentService.DetachISOFromVFlash() Method: Return Code Values.....	30
146	Table 39 – DCIM_OSDeploymentService.DetachISOFromVFlash() Method: Parameters.....	31
147	Table 40 – DCIM_OSDeploymentService.DetachISOFromVFlash() Method: Standard Messages	31
148	Table 41 – DCIM_OSDeploymentService.DeletelISOFromVFlash() Method: Return Code Values.....	31

149	Table 42 – DCIM_OSDeploymentService.DeleteISOFromVFlash() Method: Parameters.....	32
150	Table 43 – DCIM_OSDeploymentService.DeleteISOFromVFlash() Method: Standard Messages	32
151	Table 44 – DCIM_OSDeploymentService.ConnectNetworkISOImage () Method: Return Code Values ...	32
152	Table 45 – DCIM_OSDeploymentService.ConnectNetworkISOImage() Method: Parameters	33
153	Table 46 – DCIM_OSDeploymentService.ConnectNetworkISOImage () Method: Standard Messages....	33
154	Table 47 – DCIM_OSDeploymentService.DisconnectNetworkISOImage () Method: Return Code Values	34
155	Table 48 – DCIM_OSDeploymentService.DisconnectNetworkISOImage () Method: Parameters.....	34
156	Table 49 – DCIM_OSDeploymentService.DisconnectNetworkISOImage () Method: Standard Messages	34
157	Table 50 – DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo() Method: Return Code	
158	Values	35
159	Table 51 – DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo() Method: Parameters .	35
160	Table 52 – DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo () Method: Standard	
161	Messages.....	35
162	Table 53 – DCIM_OSDeploymentService.SkipISOImageBoot () Method: Return Code Values	36
163	Table 54 – DCIM_OSDeploymentService.SkipISOImageBoot() Method: Parameters	36
164	Table 55 – DCIM_OSDeploymentService.SkipISOImageBoot () Method: Standard Messages.....	36
165	Table 56 – DCIM_OSDeploymentService.DisconnectRFSISOImage() Method: Return Code Values.....	36
166	Table 57 – DCIM_OSDeploymentService.DisconnectRFSISOImage() Method: Parameters.....	37
167	Table 58 – DCIM_OSDeploymentService. DisconnectRFSISOImage() Method: Standard Messages	37
168	Table 59 – DCIM_OSDeploymentService.ConnectRFSISOImage () Method: Return Code Values	37
169	Table 60 – DCIM_OSDeploymentService.ConnectRFSISOImage() Method: Parameters	38
170	Table 61 – DCIM_OSDeploymentService.ConnectRFSISOImage() Method: Standard Messages.....	38
171	Table 62 – DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo() Method: Return Code	
172	Values	39
173	Table 63 – DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo () Method: Parameters	39
174	Table 64 – DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo() Method: Standard	
175	Messages.....	39
176	Table 65 – DCIM_OSDeploymentService.GetHostMACInfo() Method: Return Code Values.....	39
177	Table 66 – DCIM_OSDeploymentService.GetHostMACInfo() Method: Parameters.....	40
178	Table 67 – DCIM_OSDeploymentService.GetHostMACInfo() Method: Standard Messages	40
179	Table 68 – Privilege and License Requirements	40
180		

Operating System (OS) Deployment Profile

182 **1 Scope**

183 The Operating System (OS) Deployment Profile extends the management capabilities of referencing
184 profiles by adding the capability to represent OS deployment configuration features. The OS deployment
185 feature consists of:

- 186 • Provide the supported OS and version information from the embedded OS driver pack.
- 187 • Unpack the OS Driver Update Package (DUP) and extract drivers for a specified OS; the drivers
188 are placed on a local flash drive (nvram) or on a network share.
- 189 • Expose the unpacked drivers to the host as a USB device.
- 190 • Ability to boot to PXE images.
- 191 • Ability to boot to the ISO image present on a network share.
- 192 • Ability to download ISO Image and save it in vFlash.
- 193 • Ability to boot to the ISO present in vFlash.

194 **2 Normative References**

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

- 198 • DMTF DSP1033, *Profile Registration Profile 1.0.0*
- 199 • DMTF DSP0200, *CIM Operations over HTTP 1.2.0*
- 200 • DMTF DSP0004, *CIM Infrastructure Specification 2.3.0*
- 201 • DMTF DSP1000, *Management Profile Specification Template*
- 202 • DMTF DSP1001, *Management Profile Specification Usage Guide*
- 203 • *Dell Lifecycle Controller Best Practices Guide 1.0*,
204 http://en.community.dell.com/techcenter/extras/m/white_papers/20066173.aspx
- 205 • *Dell WSMAN Licenses and Privileges 1.0*
- 206 • ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
207 <http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype>
- 208 • Unified Modeling Language (UML) from the Open Management Group (OMG),
209 <http://www.uml.org>
- 210 • Dell Tech Center MOF Library: <http://www.delltechcenter.com/page/DCIM.Library.MOF>
- 211 • Related Managed Object Format (MOF) files:
 - 212 ○ DCIM_OSDeploymentService.mof

- 213 ○ DCIM_OSDConcreteJob.mof
- 214 ○ DCIM_LCElementConformsToProfile.mof
- 215 ○ DCIM_LCRegisteredProfile.mof

216 **3 Terms and Definitions**

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

218 **3.1**

219 **Conditional** – Indicates requirements to be followed strictly in order to conform to the document when the
220 specified conditions are met.

221 **3.2**

222 **Mandatory** – Indicates requirements to be followed strictly in order to conform to the document and from
223 which no deviation is permitted.

224 **3.3**

225 **May** – Indicates a course of action permissible within the limits of the document.

226 **3.4**

227 **Optional** – Indicates a course of action permissible within the limits of the document.

228 **3.5**

229 **can** – Used for statements of possibility and capability, whether material, physical, or causal.

230 **3.6**

231 **cannot** – Used for statements of possibility and capability, whether material, physical, or causal.

232 **3.7**

233 **need not** – Indicates a course of action permissible within the limits of the document.

234 **3.8**

235 **referencing profile** – Indicates a profile that owns the definition of this class and can include a reference
236 to this profile in its “Related Profiles” table.

237 **3.9**

238 **shall** – Indicates requirements to be followed strictly in order to conform to the document and from which
239 no deviation is permitted.

240 **3.10**
241 **shall not** – Indicates requirements to be followed strictly in order to conform to the document and from
242 which no deviation is permitted.

243 **3.11**
244 **should** – Indicates that among several possibilities, one is recommended as particularly suitable, without
245 mentioning or excluding others, or that a certain course of action is preferred but not necessarily required.

246 **3.12**
247 **should not** – Indicates that a certain possibility or course of action is deprecated but not prohibited

248 **3.13**
249 **FQDD** – Fully Qualified Device Descriptor is used to identify a particular component in a system.

250 **3.14**
251 **Interop Namespace** – Interop Namespace is where instrumentation instantiates classes to advertise its
252 capabilities for client discovery.

253 **3.15**
254 **Implementation Namespace** – Implementation Namespace is where instrumentation instantiates
255 classes relevant to executing core management tasks.

256 **3.16**
257 **ENUMERATE** – Refers to WS-MAN `ENUMERATE` operation as described in Section 8.2 of
258 `DSP0226_V1.1` and Section 9.1 of `DSP0227_V1.0`

259 **3.17**
260 **GET** – Refers to WS-MAN `GET` operation as defined in Section 7.3 of `DSP00226_V1.1` and Section 7.1
261 of `DSP0227_V1.0`
262

263 **4 Symbols and Abbreviated Terms**

264 **4.1**
265 **CIM**
266 Common Information Model

267 **4.2**
268 **RFS**
269 Remote File Share

270 **5 Synopsis**

271 **Profile Name:** Operating System (OS) Deployment
272 **Version:** 1.2.0
273 **Organization:** DCIM
274 **CIM Schema Version:** 2.26 Experimental
275 **Central Class:** DCIM_OSDeploymentService

276 **Scoping Class:** CIM_ComputerSystem

277 The **Operating System (OS) Deployment Profile** extends the management capability of the referencing
278 profiles by adding the capability to support OS deployment activities by manipulating the OS deployment
279 features provided by the service processor. These features include:

- 280 • Identify the list of OS drivers.
- 281 • Unpack the OS DUP and extract drivers for a specified OS; the drivers are placed on a local flash
282 drive (nvram).
- 283 • Expose the unpacked drivers to the host as a USB device.
- 284 • Expose the drivers to the network in a secure manner for access by a management application.
- 285 • Support booting to an OS provisioning image from a network share.
- 286 • Download a pre-boot OS provisioning image from a network share to vFlash.
- 287 • Support booting to an OS provisioning image from vFlash.

288 DCIM_OSDeploymentService shall be the Central Class.

289 Table 1 identifies related profiles.

290

Table 1 – Related Profiles

Profile Name	Organization	Version	Relationship
Profile Registration	DCIM	1.0	Reference

291 **6 Description**

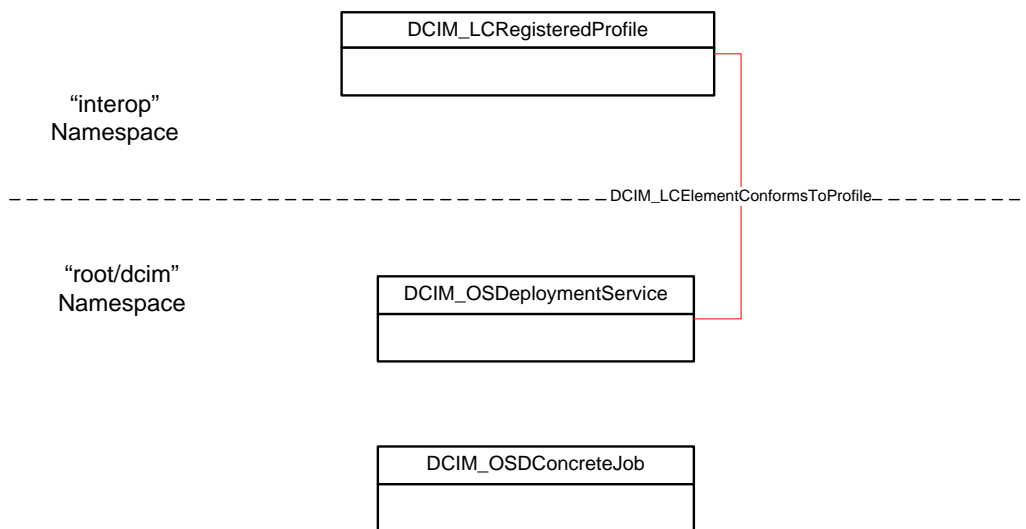
292 The Operating System (OS) Deployment Profile describes the OS deployment configuration service and
293 the related methods. The profile also describes the relationship of the OS Deployment Profile classes to
294 DMTF and Dell profile version information.

295 The OS deployment features are:

- 296 • Identifying the list of OS drivers.
- 297 • Unpacking the OS DUP and extract drivers for a specified OS. The drivers are placed on a local
298 flash drive (nvram).
- 299 • Exposing the unpacked drivers to the host as a USB device.
- 300 • Securely exposing the drivers to the network for access by a management application.
- 301 • Booting to an OS provisioning image from a network share.
- 302 • Downloading a pre-boot OS provisioning image from a network share to vFlash SD card.
- 303 • Booting to an OS provisioning image from vFlash SD card.

304 Figure 1 represents the class schema for the **Operating System (OS) Deployment Profile**. For
305 simplicity, the prefix CIM_ has been removed from the names of the classes that are standard DMTF
306 classes.

307 The OS Deployment feature in a service processor is represented by the instance of the
308 DCIM_OSDeploymentService class. The DCIM_OSDeploymentService has extrinsic methods for
309 accomplishing the various OS deployment features described above. The OS Deployment profile
310 information is represented with the instance of CIM_RegisteredProfile.



311

312

Figure 1 – Operating System (OS) Deployment Profile: Class Diagram

313 7 Implementation Requirements

314 Requirements and guidelines for propagating and formulating certain properties of the classes are
 315 discussed in this section.

316 This section describes the implementation of Dell OS Deployment.

317 **Table 2 – Class Requirements: Power State Management Profile**

Element Name	Requirement	Description
Classes		
DCIM_OSDeploymentService	Mandatory	The class shall be implemented in the “root/dcim” namespace. See section 7.1
DCIM_OSConcreteJob	Mandatory	The class shall be implemented in the “root/dcim” namespace. See section 7.2
DCIM_LCElementConformsToProfile	Mandatory	The class shall be implemented in both the <i>Interop</i> and “root/dcim” namespaces. See section 7.1 and 7.3.
DCIM_LCRegisteredProfile	Mandatory	The class shall be implemented in the “root/interop” namespace. See section 7.3.
Indications		
None defined in this profile		

318

319 7.1 DCIM_OSDeploymentService

320 This section describes the implementation for the DCIM_OSDeploymentService class.

321 This class is instantiated in the “root/dcim” namespace.

322 The DCIM_LCElementConformsToProfile association’s ManagedElement property shall references the
 323 DCIM_OSDeploymentService instance(s).

324 7.1.1 Resource URIs for WinRM®

325 The class Resource URI is:

326 *“http://schemas.dell.com/wbem/wscim/1/cim-schema/2 /*
 327 *DCIM_OSDeploymentService?__cimnamespace=root/dcim”*

328 The key properties are the SystemCreationClassName, CreationClassName, SystemName, and Name.

329 The instance Resource URI for DCIM_OSDeploymentService instance is:

330 *http://schemas.dell.com/wbem/wscim/1/cim-*
 331 *schema/2/DCIM_OSDeploymentService?__cimnamespace=root/dcim+SystemCreationClassName=*
 332 *DCIM_ComputerSystem+SystemName=DCIM:ComputerSystem+CreationClassName=DCIM_OSDe*
 333 *ploymentService +Name= DCIM:OSDeploymentService”*

334 7.1.2 Operations

335 The following table details the implemented operations on DCIM_OSDeploymentService.

336 **Table 3 –DCIM_OSDeploymentService - Operations**

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

337

338 7.1.3 Class Properties

339 The following table details the implemented properties for DCIM_OSDeploymentService instance in a
340 system. The “Requirements” column shall denote the implementation requirement for the corresponding
341 property. The “Type” column denotes the corresponding property type. The “Additional Requirement”
342 Column specifies additional information on the property value.

343 **Table 3 – DCIM_OSDeploymentService - Properties**

Property Name	Type	Requirement	Additional Requirement
CreationClassName	String	Mandatory	The property value shall be "DCIM_OSDeploymentService"
Name	String	Mandatory	The property value shall be "DCIM:OSDeploymentService"
ElementName	String	Mandatory	The property value shall be "Operating System Deployment Service"
SystemCreationClassName	String	Mandatory	The property value shall be "DCIM_ComputerSystem"
SystemName	String	Mandatory	The property value shall be "DCIM:ComputerSystem"

344

345 7.2 DCIM_OSConcreteJob

346 This section describes the implementation for the DCIM_OSConcreteJob class.

347 This class shall be instantiated in the “root/dcim” namespace.

348 7.2.1 Resource URIs for WinRM®

349 The class Resource URI is:

350 *“[http://schemas.dell.com/wbem/wscim/1/cim-schema/2 /](http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_OSConcreteJob?__cimnamespace=root/dcim)*
351 *DCIM_OSConcreteJob?__cimnamespace=root/dcim”*

352 The key property is the InstanceID property.

353 The instance Resource URI for DCIM_OSConcreteJob instance is:

354 [http://schemas.dell.com/wbem/wscim/1/cim-schema/2/](http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_OSConcreteJob?__cimnamespace=root/dcim+InstanceID=<InstanceID>)
355 [DCIM_OSConcreteJob?__cimnamespace=root/dcim+InstanceID=<InstanceID>](http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_OSConcreteJob?__cimnamespace=root/dcim+InstanceID=<InstanceID>)

356 *where <InstanceID> is the InstanceID property value.*

357 **7.2.2 Class Properties**

358 The following table lists the implemented properties for DCIM_OSDConcreteJob instance in a system.
 359 The “Requirements” column shall denote whether the property is implemented (for requirement
 360 definitions, see section 3). The “Additional Requirements” column shall denote either possible values for
 361 the property, or requirements on the value formulation.

362 **Table 4 – DCIM_OSDConcreteJob - Properties**

Property Name	Type	Requirement	Additional Requirement
InstanceID	String	Mandatory	The property shall have unique and opaque value.
JobStatus	String	Mandatory	The property shall have a value from the “Job Status” column in the Table 5.
JobName	String	Mandatory	The property shall have a value from the “Job Name” column in the Table 5.
DeleteOnCompletion	Boolean	Mandatory	The property shall denote whether the DCIM_OSDConcreteJob instance will be deleted on the completion. The value of TRUE shall represent the deletion on completion.
Message	String	Mandatory	The property shall have a message string from the Dell Message Registry.
MessageID	String	Mandatory	The property shall have a message ID from the Dell Message Registry.
MessageArgument[]	String	Optional	

363 The following table lists the JobName and JobStatus property values, and their respective descriptions.

364 **Table 5 – Job Descriptions**

Job Name	Job Status	JobStatus Description
UnpackAndAttach	Processing Driver Pack	Extracting drivers, creating dynamic partition, copying drivers, and attaching the partition as a USB device to the host.
	Success	Successfully executed the method.
	Failed	Failed to execute the method. For more information, see MessageID and Message properties on the returned DCIM_OSDConcreteJob instance.
	Processing Driver Pack	Extracting drivers and copying drivers to the network share.
	Success	Successfully executed the method.
	Failed	Failed to execute the method. For more information, see MessageID and Message properties on the returned DCIM_OSDConcreteJob instance.
BootToNetworkISO	Rebooting to ISO	Attaching the network ISO image as a local CD-ROM to the host and booting to it.
	Success	Successfully executed the method.
	Failed	Failed to execute the method, For more information, see MessageID and Message properties on the returned DCIM_OSDConcreteJob instance.
BootToISOFromVFlash	Rebooting to ISO	Attaching the ISO image on vFlash as a local

Job Name	Job Status	JobStatus Description
		CD-ROM to the host and booting to it.
	Success	Successfully executed the method.
	Failed	Failed to execute the method, refer to MessageID and Message properties on the returned DCIM_OSDConcreteJob instance for detailed information.
DownloadISOToVFlash	Downloading	Copying the ISO image from network share to vFlash SD card.
	Success	Successfully executed the method.
	Failed	Failed to execute the method. For more information, see MessageID and Message properties on the returned DCIM_OSDConcreteJob instance.
ConnectNetworkISOImage	Connecting to Network ISO	Connecting to the ISO present on a network share and attaching it as a CD-ROM device to the host.
	Success	Successfully executed the method.
	Failed	Failed to execute the method. For more information, see MessageID and Message properties on the returned DCIM_OSDConcreteJob instance.
ConnectRFSISOImage	Connecting to Network ISO	Connecting to the ISO present on a network share, and attaching it to the host server as a USB CD-ROM device through RFS end point.
	Success	Successfully executed the method.
	Failed	Failed to execute the method. For more information, see MessageID and Message properties on the returned DCIM_OSDConcreteJob instance.

365 **7.3 Operating System (OS) Deployment Profile Registration –** 366 **DCIM_LCRegistered Profile**

367 This section describes the implementation for the DCIM_LCRegisteredProfile class.

368 This class shall be instantiated in the “root/interop” namespace.

369 The DCIM_LCElementConformsToProfile association(s)' ConformantStandard property shall reference
370 the DCIM_LCRegisteredProfile instance.

371 **7.3.1 Resource URIs for WinRM®**

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

374 The key property shall be the InstanceID property.

375 The instance WBEM URI shall be: “http://schemas.dell.com/wbem/wscim/1/cim-
376 schema/2/DCIM_LCRegisteredProfile?__cimnamespace=root/interop+InstanceID=
377 DCIM:OSDeployment:1.1.0”

378 **7.3.2 Operations**

379 The following table details the implemented operations on DCIM_LCRegisteredProfile.

380

Table 6 – DCIM_LCRegisteredProfile - Operations

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

381

382 7.3.3 Class Properties

383 The following table lists the implemented properties for DCIM_LCRegisteredProfile instance representing
 384 Operating System (OS) Deployment Profile implementation. The “Requirements” column shall denote
 385 whether the property is implemented (for requirement definitions, see section 3). The “Additional
 386 Requirements” column shall denote either possible values for the property, or requirements on the value
 387 formulation.

388

Table 7 – Class: DCIM_LCRegisteredProfile

Properties	Type	Requirement	Additional Requirements
InstanceID	string	Mandatory	This property shall have a value of “DCIM:OSDeployment:1.1.0”
RegisteredName	string	Mandatory	This property shall have a value of “OS Deployment”.
RegisteredVersion	string	Mandatory	This property shall have a value of “1.4.0”.
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 “root/interop” namespace”
AdvertisedTypes[]	uint16	Mandatory	This property array shall contain [1(Other), 1(Other)].
AdvertiseTypeDescriptions[]	string	Mandatory	This property array shall contain [“WS-Identify”, “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.

389

390 **8 Methods**

391 This section details the requirements for supporting extrinsic methods for the CIM elements defined by
392 this profile.

393 **8.1 DCIM_OSDeploymentService.GetDriverPackInfo()**

394 The GetDriverPackInfo() method is used get the list of operating systems that can be installed on the host
395 system using the embedded device drivers present in the Lifecycle Controller.

396 The following table lists the return values for GetDriverPackInfo() method, where the method-execution
397 behavior matches the return-code description.

398 **Table 8 – DCIM_OSDeploymentService.GetDriverPackInfo() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

399 The following table lists the output parameters for GetDriverPackInfo() method.

400

Table 9 – DCIM_OSDeploymentService.GetDriverPackInfo() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT, REQ	Version	String	NULL or version of the driver pack.
OUT, REQ	OSList[]	String	NULL or contains the list of operating systems supported for this server.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages.

401

Table 10 – DCIM_OSDeploymentService.GetDriverPackInfo() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD2	General failure	
OSD3	Lifecycle Controller is being used by another process	
OSD4	Cannot access Driver Pack partition in Lifecycle Controller	
OSD5	Driver Pack not found in Lifecycle Controller	
OSD6	Cannot allocate memory	
OSD29	Driver Pack config file not found in Lifecycle Controller. Driver Pack might be corrupt	
OSD35	Lifecycle Controller is disabled	

402

8.2 DCIM_OSDeploymentService.UnpackAndAttach()

403

The UnpackAndAttach() method is used to extract the drivers for the selected operating system to a USB device that is attached locally to the server for the specified time interval. The following table specifies the return values for UnpackAndAttach() method, where the method-execution behavior matches the return-code description.

404

405

406

407

Table 11 – DCIM_OSDeploymentService.UnpackAndAttach() Method: Return Code Values

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

408

The following table lists the output parameters for UnpackAndAttach() method.

Table 12 – DCIM_OSDeploymentService.UnpackAndAttach() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	OSName	String	Name of the OS to unpack drivers for, this value shall match one of the strings in OSList returned for GetDriverPackInfo
IN, REQ	ExposeDuration	DateTime	Identifies the amount of time up to 18 hours for the drivers to be exposed as an USB device to the host.
OUT, REQ	Job	CIM_ConcreteJob REF	Returned to keep track of OSD – USB attach job.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages

Table 13 – DCIM_OSDeploymentService.UnpackAndAttach() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD1	The command was successful	
OSD2	General failure	
OSD3	Lifecycle Controller is being used by another process	
OSD4	Cannot access Driver Pack partition in Lifecycle Controller	
OSD5	Driver Pack not found in Lifecycle Controller	
OSD10	Installation not supported for the selected operating system	
OSD11	Driver Pack does not have drivers for the selected operating system	
OSD12	Cannot create USB device to copy drivers for the selected operating system	
OSD13	Cannot mount USB device to copy drivers for the selected operating system	
OSD19	The fork() command for a child process to do the task failed	
OSD20	Unable to get size or label from Driver Pack for selected operating system	
OSD27	Copying drivers for selected operating system failed	
OSD30	Invalid value for ExposeDuration – must be 60-65535 seconds	
OSD35	Lifecycle Controller is not enabled	

412 **8.3 DCIM_OSDeploymentService.DetachDrivers()**

413 This OSDeploymentService.DetachDrivers() method is used to detach the USB device containing the
 414 drivers from the host server. The following table lists the return values for DetachDrivers() method,, where
 415 the method-execution behavior matches the return-code description.
 416

417 **Table 14 – DCIM_OSDeploymentService.DetachDrivers() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

418 The following table lists the output parameters for the DetachDrivers() method.

419 **Table 15 – DCIM_OSDeploymentService.DetachDrivers() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages

420 **Table 16 – DCIM_OSDeploymentService.DetachDrivers() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
OSD7	Unable to retrieve Lifecycle Controller handle	
OSD24	Unable to continue with DetachDrivers - UnPackAndAttach is in progress	
OSD25	Unable to detach USB device containing operating system drivers	

421 **8.4 DCIM_OSDeploymentService.UnpackAndShare()**

423 The UnpackAndShare() method is used to extract the drivers for the selected operating system, and copy
 424 them to the specified network share. The following table lists the return values for UnpackAndShare(),
 425 where the method-execution behavior matches the return-code description.

426 **Table 17 – DCIM_OSDeploymentService.UnpackAndShare() Method: Return Code Values**

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

427 The following table lists the input and output parameters for the UnpackAndShare() method.

Table 18 – DCIM_OSDeploymentService.UnpackAndShare() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	String	CIFS or NFS share IPv4 address. For example, 192.168.10.100
IN, REQ	ShareName	String	NFS or CIFS share name. For example, "/home/guest" or "guest_smb" respectively.
IN, REQ	OSName	String	Operating System name
IN, REQ	ShareType	UInt32	0 = NFS or CIFS = 2
IN, REQ	Workgroup	String	Workgroup name, if applicable.
IN	UserName	String	User name, if applicable.
IN	Password	String	Password, if applicable.
IN	Port	UInt32	Port number, if applicable.
OUT, REQ	Job	CIM_ConcreteJob REF	Returned to keep track of OSD – UnpackAndShare job.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages.

Table 19 – DCIM_OSDeploymentService.UnpackAndShare() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD1	The command was successful	
OSD2	General failure	
OSD3	Lifecycle Controller is being used by another process	
OSD4	Cannot access Driver Pack partition in Lifecycle Controller	
OSD5	Driver Pack not found in Lifecycle Controller	
OSD10	Installation not supported for the selected operating system	
OSD11	Driver Pack does not have drivers for the selected operating system	
OSD16	Mount network share failed - incorrect IP address or share name	
OSD19	The fork() command for a child process to do the task failed	
OSD20	Unable to get size or label from Driver Pack for selected operating system	
OSD31	Copying operating system drivers to network share failed	
OSD35	Lifecycle Controller is not enabled	
OSD47	Inaccessible network share	

431 **8.5 DCIM_OSDeploymentService.BootToNetworkISO()**

432 The BootToNetworkISO() method is used to boot to the iDRAC pre-operating system image that was
433 already downloaded. The following table lists the return values for BootToNetworkISO() method, where
434 the method-execution behavior matches the return-code description.

435 **Table 20 – DCIM_OSDeploymentService.BootToNetworkISO() Method: Return Code Values**

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

436 The following table lists the input and output parameters for the BootToNetworkISO() method.

Table 21 – DCIM_OSDeploymentService.BootToNetworkISO() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	String	NFS or CIFS share IPv4 address. For example, 192.168.10.100
IN, REQ	ShareName	String	NFS or CIFS share name. For example, "/home/guest" or "guest_smb" respectively.
IN, REQ	ImageName	String	ISO image name
IN, REQ	ShareType	Uint32	0=NFS or CIFS = 2
IN, REQ	Workgroup	String	Workgroup name, if applicable.
IN	UserName	String	User name, if applicable.
IN	Password	String	Password, if applicable.
IN	Port	Uint32	Port number, if applicable.
IN	ExposeDuration	DateTime	Identifies the amount of time (up to 18 hours) for the ISO Image to be exposed as a local CD-ROM device to the host.
IN	HashType	Uint16	Type of hash algorithm used to compute checksum (1=MD5 or 2=SHA1)
IN	HashValue	String	Checksum value in string format computed using the HashType algorithm
OUT, REQ	Job	CIM_ConcreteJob REF	Returned to keep track of OSD – UnpackAndShare job.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages

Table 22 – DCIM_OSDeploymentService.BootToNetworkISO() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD1	The command was successful	
OSD3	Lifecycle Controller is being used by another process	
OSD9	Failed to reboot the system using IPMI command	
OSD16	Mount network share failed - incorrect IP address or share name	
OSD17	Exposing ISO image as internal device to the host system failed	
OSD19	The fork() command for a child process to do the task failed	
OSD21	Unable to boot to ISO image	
OSD28	Hash verification on the ISO image failed	
OSD30	Invalid value for ExposeDuration - must be 60-65535 seconds	
OSD35	Lifecycle Controller is not enabled	
OSD36	Boot to ISO Image has been	

MessageID (OUT parameter)	Message	MessageArguments[]
	cancelled by user using CTRL+E option on the server	
OSD47	Inaccessible network share	
OSD50	Lifecycle Controller is in field service mode	
OSD51	Reboot the system to run pending Lifecycle Controller Tasks	

439

440 8.6 DCIM_OSDeploymentService.DetachISOImage()

441 The DetachISOImage() method is used to detach the ISO Image from the host server. The following table
 442 lists the return values for DetachISOImage() method, where the method-execution behavior matches the
 443 return-code description. The following table lists the return values for BootToNetworkISO() method, where
 444 the method-execution behavior matches the return-code description.
 445

446 **Table 23 – DCIM_OSDeploymentService.DetachISOImage() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

447 The following table lists the output parameters for the UnpackAndShare() method.

448

Table 24 – DCIM_OSDeploymentService.DetachISOImage() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	String	Substitution variables for dynamic error messages

449

Table 25 – DCIM_OSDeploymentService.DetachISOImage() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD7	Unable to retrieve Lifecycle Controller handle	
OSD22	Unable to detach ISO image from the host	
OSD23	Unable to continue with DetachISOImage - another command is in the process of exposing ISO Image and booting to it	
OSD32	ISO image is not attached	

450

451 **8.7 DCIM_OSDeploymentService.BootToPXE()**

452 The BootToPXE() method is used to boot a host system using the PXE mechanism. The following table
 453 lists the return values for the BootToPXE() method, where the method-execution behavior matches the
 454 return-code description.

455

Table 26 – DCIM_OSDeploymentService.BootToPXE() Method: Return Code Values

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

456

The following table lists the output parameters for the BootToPXE() method.

457

Table 27 – DCIM_OSDeploymentService.BootToPXE() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages

458

Table 28 – DCIM_OSDeploymentService.BootToPXE() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD3	Lifecycle Controller is being used by another process	
OSD8	Setting Boot to PXE through IPMI failed	
OSD9	Failed to reboot the system using IPMI command	
OSD26	Unable to continue with BootToPXE - another command is running	
OSD34	Unable to continue with BootToPXE - ISO image is attached to the system	
OSD35	Lifecycle Controller is disabled	

459

460

461

8.8 Method: DCIM_OSDeploymentService.BootToHD()

462

The BootToHD() method is used for one time boot to the host server's hard disk. Return values for BootToHD() shall be as specified in Table 29, where the method-execution behavior matches the return-code description. BootToHD() method's parameters are specified in Table 30.

463

464

465

Table 29 – DCIM_OSDeploymentService.BootToHD() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred.

466

Table 30 – DCIM_OSDeploymentService.BootToHD() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages

467

468

Table 31 – DCIM_OSDeploymentService.BootToHD() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD3	Lifecycle Controller is being used by another process	
OSD8	Setting Boot to PXE through IPMI failed	
OSD9	Failed to reboot the system using IPMI command	
OSD35	Lifecycle Controller is disabled	

469

470

471

472

8.9 DCIM_OSDeploymentService.DownloadISOToVFlash()

473

The DownloadISOToVFlash() method is used to download the pre-operating system ISO Image to the vFlash SD card. The following table lists the return values for DownloadISOToVFlash() method, where the method-execution behavior matches the return-code description.

474

475

476

Table 32 – DCIM_OSDeploymentService.DownloadISOToVFlash() Method: Return Code Values

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

477

478

The following table list the input and output parameters for the DownloadISOToVFlash() method.

Table 33 – DCIM_OSDeploymentService.DownloadISOToVFlash() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	String	TFTP, CIFS, or NFS share IPv4 address. For example, 192.168.10.100
IN, REQ	ShareName	String	NFS or CIFS network share point. For example, "/home/guest" or "guest_smb."
IN, REQ	ImageName	String	ISO Image name
IN, REQ	ShareType	UInt32	<ul style="list-style-type: none"> • 0 (NFS) • 1 (TFTP) • 2 (CIFS)
IN	Workgroup	String	Workgroup name, if applicable.
IN	UserName	String	User name, if applicable.
IN	Password	String	Password, if applicable.
IN	Port	UInt32	Port number, if applicable.
IN	HashType	UInt16	Type of hash algorithm used to compute checksum: <ul style="list-style-type: none"> • 1 (MD5) • 2 (SHA1)
IN	HashValue	String	Checksum value in string format computed using HashType algorithm.
OUT, REQ	Job	CIM_ConcreteJob REF	Returned to keep track of OSD – UnpackAndShare job.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages

Table 34 – DCIM_OSDeploymentService.DownloadISOToVFlash() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD1	The command was successful	
OSD3	Lifecycle Controller is being used by another process	
OSD16	Mount network share failed - incorrect IP address or share name	
OSD18	Unable to locate the ISO image on the network share point	
OSD19	The fork() command for a child process to do the task failed	
OSD28	Hash verification on the ISO image failed	

MessageID (OUT parameter)	Message	MessageArguments[]
OSD37	ISO image size too large	
OSD40	VFlash is not Dell-licensed	
OSD42	Downloading ISO File to VFlash failed	
OSD43	VFlash unavailable	
OSD47	Inaccessible network share	
OSD48	ISO Image more than 4GB not supported	
OSD52	VFlash is Disabled	
OSD53	VFlash is write-protected	
OSD54	VFlash already has 16 partitions	
OSD55	ISO Image is attached to host	

481

482 8.10 DCIM_OSDeploymentService.BootToISOFromVFlash()

483 The BootToISOFromVFlash() method is used to boot to the downloaded pre-operating system image on
 484 the vFlash SD card. The following table lists the return values for BootToISOFromVFlash() method, where
 485 the method-execution behavior matches the return-code description.

486 **Table 35 – DCIM_OSDeploymentService.BootToISOFromVFlash() Method: Return Code Values**

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

487 The following table list the BootToISOFromVFlash() method.

488 **Table 316 – DCIM_OSDeploymentService.BootToISOFromVFlash() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT, REQ	Job	CIM_ConcreteJob REF	Returned to keep track of the created job.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages

489 **Table 37 – DCIM_OSDeploymentService.BootToISOFromVFlash() Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
OSD1	The command was successful	
OSD3	Lifecycle Controller is being used by another process	
OSD9	Failed to reboot the system using IPMI command	

MessageID (OUT parameter)	Message	MessageArguments[]
OSD19	The fork() command for a child process to do the task failed	
OSD21	Unable to boot to ISO image	
OSD36	Boot to ISO Image has been cancelled by user using CTRL+E option on the server	
OSD40	VFlash is not Dell-licensed	
OSD41	ISO Image not found on VFlash	
OSD43	VFlash unavailable	
OSD50	Lifecycle Controller is in field service mode	
OSD51	Reboot the system to run pending Lifecycle Controller Tasks	
OSD52	VFlash is Disabled	
OSD53	VFlash is write-protected	
OSD55	ISO Image is attached to host	

490

491 **8.11 DCIM_OSDeploymentService.DetachISOFromVFlash()**

492 The DetachISOFromVFlash() method is used to detach the ISO Image (on a vFlash SD card) from the
493 host system.

494 The following table returns the values for DetachISOFromVFlash(), where the method-execution behavior
495 matches the return-code description.
496

497 **Table 38 – DCIM_OSDeploymentService.DetachISOFromVFlash() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

498 The following table lists the parameters for the DetachISOFromVFlash() method.

499

Table 39 – DCIM_OSDeploymentService.DetachISOFromVFlash() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages.

500

Table 40 – DCIM_OSDeploymentService.DetachISOFromVFlash() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD23	Unable to continue with DetachISOImage - another command is in the process of exposing ISO Image and booting to it	
OSD39	Unable to find the VFlash	
OSD40	VFlash is not Dell-licensed	
OSD41	ISO Image not found on Vflash	
OSD43	Vflash unavailable	
OSD44	Unable to detach ISO image on Vflash	
OSD52	VFlash is Disabled	
OSD53	VFlash is write-protected	

501

502

503

8.12 DCIM_OSDeploymentService.DeleteISOFromVFlash()

504

The DeleteISOFromVFlash() method is used to delete the ISO Image from vFlash SD card.

505

The following table lists the return values for the DeleteISOFromVFlash() method, where the method-execution behavior matches the return-code description.

506

507

Table 41 – DCIM_OSDeploymentService.DeleteISOFromVFlash() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred.

508

The following table lists the parameters for the DeleteISOFromVFlash().

510

Table 42 – DCIM_OSDeploymentService.DeleteISOFromVFlash() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	String	Substitution variables for dynamic error messages

511

Table 43 – DCIM_OSDeploymentService.DeleteISOFromVFlash() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD3	Lifecycle Controller is being used by another process	
OSD39	Unable to find the VFlash	
OSD40	VFlash is not Dell-licensed	
OSD41	ISO Image not found on Vflash	
OSD43	Vflash unavailable	
OSD45	Cannot delete ISO image from Vflash	
OSD52	Vflash is Disabled	
OSD53	Vflash is write-protected	

512

513

8.13 DCIM_OSDeploymentService.ConnectNetworkISOImage()

514

515 The ConnectNetworkISOImage() method is used to connect to the ISO present on the network share and
 516 expose the ISO as a local USB CD-ROM device to the host system. This method will connect to the ISO
 517 located on an NFS/CIFS share and expose it as a virtual CDROM device to the host server. Even though
 518 the successful method execution shall not change the boot order of that device, the host shall always
 519 boot to the virtual CDROM. Also note that after the ISO is exposed to the host server, Lifecycle Controller
 520 shall be locked and no other jobs like configuration/update etc shall be performed until the ISO is
 521 detached using the DisconnectNetworkISOImage() method. The successful execution of the
 522 DisconnectNetworkISOImage() shall revert the host system to the regular boot list.

523

524 The following table lists the return values for the ConnectNetworkISOImage() method, where the method-
 525 execution behavior matches the return-code description.

526

NOTE: The recommended methodology for connecting to an ISO image is by using the
 527 ConnectRFSISOImage() method that utilizes the remote file system (RFS). See section 0 for more details.

528

Table 44 – DCIM_OSDeploymentService.ConnectNetworkISOImage () Method: Return Code Values

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

529

The following table lists the input and output parameters for the ConnectNetworkISOImage() method.

Table 45 – DCIM_OSDeploymentService.ConnectNetworkISOImage() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	String	CIFS or NFS share IPv4 address. For example, 192.168.10.100
IN, REQ	ShareName	String	NFS or CIFS network share point. For example, "/home/guest" or "guest_smb."
IN, REQ	ImageName	String	ISO image name
IN, REQ	ShareType	UInt32	<ul style="list-style-type: none"> • 0 (NFS) • 2 (CIFS)
IN	Workgroup	String	Workgroup name, if applicable
IN	UserName	String	User name, if applicable.
IN	Password	String	Password, if applicable.
IN	HashType	UInt16	Type of hash algorithm used to compute checksum: <ul style="list-style-type: none"> • 1 (MD5) • 2(SHA1)
IN	HashValue	String	Checksum value in string format computed using HashType algorithm
IN	AutoConnect	Boolean	Auto-connect to ISO image up on iDRAC reset
OUT, REQ	Job	CIM_ConcreteJob REF	Returned to keep track of OSD – UnpackAndShare job.
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	String	Substitution variables for dynamic error messages

531 **Table 46 – DCIM_OSDeploymentService.ConnectNetworkISOImage () Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
OSD1	The command was successful	
OSD3	Lifecycle Controller is being used by another process	
OSD16	Mount network share failed - incorrect IP address or share name	
OSD17	Exposing ISO image as internal device to the host system failed	
OSD19	The fork() command for a child process to do the task failed	
OSD28	Hash verification on the ISO image failed	
OSD35	Lifecycle Controller is not enabled	
OSD47	Inaccessible network share	
OSD50	Lifecycle Controller is in field service mode	
OSD51	Reboot the system to run pending Lifecycle Controller Tasks	

MessageID (OUT parameter)	Message	MessageArguments[]
OSD61	IP Address is Invalid	

532

533 8.14 DCIM_OSDeploymentService.DisconnectNetworkISOImage()

534

535 The DisconnectNetworkISOImage() method is used to disconnect and detach the ISO Image from the
 536 host system. The following table lists the return values for the DisconnectNetworkISOImage() method,
 537 where the method-execution behavior matches the return-code description.

538 **Table 47 – DCIM_OSDeploymentService.DisconnectNetworkISOImage () Method: Return Code**
 539 **Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

540 The following table lists the output parameters for the DisconnectNetworkISOImage().

541 **Table 48 – DCIM_OSDeploymentService.DisconnectNetworkISOImage () Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	String	Substitution variables for dynamic error messages

542 **Table 49 – DCIM_OSDeploymentService.DisconnectNetworkISOImage () Method: Standard**
 543 **Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
OSD7	Unable to retrieve Lifecycle Controller handle	
OSD22	Unable to detach ISO image from the host	
OSD32	ISO image is not attached	

544

545

546 8.15 DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo()

547 The GetNetworkISOImageConnectionInfo() method is used to give the status of the ISO Image that has
 548 been exposed to the host system. The following table lists the return values for
 549 GetNetworkISOImageConnectionInfo(), where the method-execution behavior matches the return-code
 550 description.

551

552
553

Table 50 – DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo() Method: Return Code Values

Value	Description
0	Request was successfully executed.
2	Error occurred.

554

555 The following table lists the the output parameters for the GetNetworkISOImageConnectionInfo() method.

556
557

Table 51 – DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT, REQ	IPAddress	String	CIFS, or NFS share IPv4 address. Example 192.168.10.100
OUT, REQ	ShareName	String	Network share point. Here is an example for share name. NFS share name - "/home/guest". CIFS share name - "guest_smb"
OUT, REQ	ImageName	String	ISO Image name
OUT, REQ	ShareType	uint32	<ul style="list-style-type: none"> • 0 (NFS) • 2 (CIFS)
OUT	Workgroup	String	Workgroup name, if applicable
OUT, REQ	ISOConnectionStatus	UInt8	Describes if the ISO connection status. If the ISO is still accessible or not.
OUT, REQ	HostAttachedStatus	UInt8	Describes ISO attached status. If the ISO is attached to the host server or not.
OUT, REQ	HostBootedFromISO	UInt8	Describes ISO boot status. If the ISO has been booted atleast once or not
OUT	UserName	String	User name, if applicable
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	String	Substitution variables for dynamic error messages

558
559

Table 52 – DCIM_OSDeploymentService.GetNetworkISOImageConnectionInfo () Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD32	ISO image is not attached	

560

561 **8.16 DCIM_OSDeploymentService.SkipISOImageBoot()**

562 The SkipISOImageBoot() method allows the BIOS to skip booting to the ISO once and boot normally
563 (boot to the first device in boot list). The following table lists the return values for the SkipISOImageBoot()
564 method, where the method-execution behavior matches the return-code description.

565 **Table 53 – DCIM_OSDeploymentService.SkipISOImageBoot () Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

566 The following table lists the output parameters for the SkipISOImageBoot() method.

567 **Table 54 – DCIM_OSDeploymentService.SkipISOImageBoot() Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages

568 **Table 55 – DCIM_OSDeploymentService.SkipISOImageBoot () Method: Standard Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
OSD32	ISO image is not attached	

569 **8.17 DCIM_OSDeploymentService.DisconnectRFSISOImage()**

570 The DisconnectRFSISOImage() method is used to disconnect and detach the ISO Image that is mounted
 571 through Remote File Share (RFS) and is exposed to the host system as a USB-based CD-ROM device.
 572 The following table returns the values for the DisconnectRFSISOImage(), where the method-execution
 573 behavior matches the return-code description.

574 **Table 56 – DCIM_OSDeploymentService.DisconnectRFSISOImage() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

575

576 The following table lists the output parameters for the DisconnectRFSISOImage() method.

577

Table 57 – DCIM_OSDeploymentService.DisconnectRFSISOImage() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT, REQ	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT, REQ	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments[]	string	Substitution variables for dynamic error messages

578

Table 58 – DCIM_OSDeploymentService. DisconnectRFSISOImage() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD59	Cannot disconnect from ISO using RFS	
OSD60	ISO is not connected using RFS	

579

8.18 DCIM_OSDeploymentService.ConnectRFSISOImage()

580 The ConnectRFSISOImage() method is used to connect the ISO image that is mounted through Remote
 581 File Share (RFS) and is exposed to the host system as a USB-based CD-ROM device. The successful
 582 execution of this method shall connect to the ISO located on NFS/CIFS share to the host server and
 583 expose it as a virtual CDROM device using RFS USB endpoint. The successful execution of the method
 584 shall not change the boot order of that device. In order to boot to the CD-ROM, the CD-ROM shall be
 585 configured in the boot order in a separate step (using BIOS and Boot Management Profile), and the host
 586 server shall boot to the CD-ROM. Unlike the ConnectNetworkISOImage() method, the Lifecycle Controller
 587 is not locked and may perform other management tasks.
 588

589

590 The following table lists the return values for the ConnectRFSISOImage() method, where the method
 591 execution behavior matches the return-code description.

592

Table 59 – DCIM_OSDeploymentService.ConnectRFSISOImage () Method: Return Code Values

Value	Description
2	Error occurred.
4096	Job Created

593

594 The following table lists the parameters ConnectRFSISOImage().

Table 60 – DCIM_OSDeploymentService.ConnectRFSISOImage() Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	IPAddress	String	IP address of the host system that hosts the ISO Image
IN, REQ	ShareName	String	Share name of the host system that hosts the ISO Image
IN, REQ	ImageName	string	ISO Image name on the host system.
IN, REQ	ShareType	Uint16	Type of the share: <ul style="list-style-type: none"> • 0 (NFS) • 2 (CIFS)
IN	Username	string	User name of the account to access the network share.
IN	Password	string	Password of the account to access the network share.
IN	Workgroup	string	Workgroup of the account to access the share
IN	HashType	uint16	Type of Hash algorithm used to compute checksum: <ul style="list-style-type: none"> • 1 (MD5) • 2(SHA1)
IN, REQ	HashValue	string	Checksum value in string format computed using HashType algorithm.
OUT, REQ	Job	CIM_ConcreteJob REF	Reference to a CIM_ConcreteJob
OUT	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages.

Table 61 – DCIM_OSDeploymentService.ConnectRFSISOImage() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD16	Mount network share failed - incorrect IP address or share name	
OSD18	Unable to locate the ISO image on the network share point	
OSD19	The fork() command for a child process to do the task failed	
OSD28	Hash verification on the ISO image failed	
OSD47	Inaccessible network share	
OSD56	An ISO is already connected using RFS	
OSD57	RFS is in auto-attach mode. iDRAC attribute AttachMode value needs to be changed to Attach	
OSD58	Cannot connect to ISO using RFS	

597 **8.19 DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo()**

598 The GetRFSISOImageConnectionInfo() method is used to provide the status of the ISO Image
 599 connection that has been exposed to the host system. The following method returns the values for
 600 GetRFSISOImageConnectionInfo() method, where the method-execution behavior matches the return-
 601 code description.

602 **Table 62 – DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo() Method: Return Code**
 603 **Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

604 The following table lists the output parameters for the GetRFSISOImageConnectionInfo().

605 **Table 63 – DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo () Method: Parameters**

Qualifiers	Name	Type	Description/Values
OUT, REQ	FilePath	String	File Path of ISO Image in the form of URI that includes IP address, share name and Image name
OUT	UserName	String	User Name as specified in the ConnectNetworkISOImage method
OUT, REQ	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT, REQ	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	string	Substitution variables for dynamic error messages.

606 **Table 32 – DCIM_OSDeploymentService.GetRFSISOImageConnectionInfo() Method: Standard**
 607 **Messages**

MessageID (OUT parameter)	Message	MessageArguments[]
OSD60	ISO is not connected using RFS	

608 **8.20 DCIM_OSDeploymentService.GetHostMACInfo()**

609 The GetHostMACInfo() method is used to provide the list of mac addresses from the system.

610 **Table 65 – DCIM_OSDeploymentService.GetHostMACInfo() Method: Return Code Values**

Value	Description
0	Request was successfully executed.
2	Error occurred.

611 The following table lists the output parameters for the GetRFSISOImageConnectionInfo().

612

Table 66 – DCIM_OSDeploymentService.GetHostMACInfo() Method: Parameters

Qualifiers	Name	Type	Description/Values
OUT, REQ	MACList	String	Contains the list of mac addresses from the system
OUT, REQ	MessageID	String	If the method fails to execute, the error message ID is returned.
OUT, REQ	Message	String	If the method fails to execute, the error message in English is returned.
OUT	MessageArguments []	String	Substitution variables for dynamic error messages.

613

Table 67 – DCIM_OSDeploymentService.GetHostMACInfo() Method: Standard Messages

MessageID (OUT parameter)	Message	MessageArguments[]
OSD2	General failure	

614 9 Use Cases

615 See *Lifecycle Controller (LC) Integration Best Practices Guide*.

616 10 CIM Elements

617 No additional requirements.

618 11 Privilege and License Requirement

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

622

Table 68 – Privilege and License Requirements

Class and Method	Operation	User Privilege Required	License Required
DCIM_OSDeploymentService	ENUMERATE, GET	Login	None.
DCIM_OSDeploymentService. GetHostMACInfo()	INVOKE	Login	None.
DCIM_OSDeploymentService. GetDriverPackInfo()	INVOKE	Login	LM_REMOTE_OS_DEPLOY MENT
DCIM_OSDeploymentService. UnpackAndAttach()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOY MENT
DCIM_OSDeploymentService. DetachDrivers()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOY MENT
DCIM_OSDeploymentService. DetachISOImage()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOY MENT

Class and Method	Operation	User Privilege Required	License Required
DCIM_OSDeploymentService. UnpackAndShare()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. BootToNetworkISO()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. BootToPXE()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. DownloadISOToVFlash()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. BootToISOFromVFlash()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. DetachISOFromVFlash()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. DeleteISOFromVFlash()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. ConnectNetworkISOImage()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. DisconnectNetworkISOImage()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. GetNetworkISOImageConnectionInfo()	INVOKE	Login	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. SkipISOImageBoot()	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. ConnectRFSISOImage	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. DisconnectRFSISOImage	INVOKE	Login, System Control	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. GetRFSISOImageConnectionInfo	INVOKE	Login	LM_REMOTE_OS_DEPLOYMENT
DCIM_OSDeploymentService. DeleteISOFromVFlash	INVOKE	Login	LM_REMOTE_OS_DEPLOYMENT
DCIM OSDConcreteJob	ENUMERATE, GET	Login	None.
DCIM_LCRegisteredProfile	ENUMERATE, GET	Login	None.
DCIM_LCElementConformsToProfile	ENUMERATE, GET	Login	None.

623

624