Dell® BIOS and Boot Management Profile

Document Number: DCIM1040
Document Type: Specification
Document Status: Published
Document Language: E
Date: 2011-03-22

Version: 1.1.0

THIS PROFILE IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHBIOSAL 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.

CONTENTS

Figures

Tables

BIOS and Boot Management Profile

1 Scope

The BIOS and Boot Management Profile extends the management capabilities of referencing profiles by adding the capability to represent the configuration of the system BIOS setup and to manage the boot of the system. The system BIOS setup is modeled with multiple attributes that allow configuration of the BIOS.

2 Normative References

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

2.1 Approved References

DMTF DSP1033, *Profile Registration Profile 1.0.0*

DMTF DSP1061, *Management Profile 1.0.0*

DMTF DSP0200, *CIM Operations over HTTP 1.2.0*

DMTF DSP0004, *CIM Infrastructure Specification 2.3.0*

DMTF DSP1000, *Management Profile Specification Template*

DMTF DSP1001, *Management Profile Specification Usage Guide*

DMTF DSP0226, *Web Services for Management (WS-Management) Specification 1.1.0*

DMTF DSP0227, *WS-Management CIM Binding Specification 1.0.0*

2.2 Other References

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

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

BIOS Boot Specification v1.01 (January 11, 1996), http://www.phoenix.com/NR/rdonlyres/56E38DE2-3E6F-4743-835F-B4A53726ABED/0/specsbbs101.pdf

3 Terms and Definitions

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

3.1

can

used for statements of possibility and capability, whether material, physical, or causal

3.2

cannot

used for statements of possibility and capability, whether material, physical, or causal

3.3

conditional

indicates requirements to be followed strictly in order to conform to the document when the specified conditions are met

3.4

mandatory

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

3.5

may

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

3.6

need not

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

3.7

optional

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

3.8

referencing profile

indicates a profile that owns the definition of this class and can include a reference to this profile in its “Related Profiles” table

3.9

shall

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

3.10

shall not

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

3.11

should

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

3.12

should not

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

3.13

`ENUMERATE`

Refers to WS-MAN `ENUMERATE` operation as described in Section 8.2 of DSP0226_V1.1 and Section 9.1 of DSP0227_V1.0

3.14

`GET`

Refers to WS-MAN `GET` operation as defined in Section 7.3 of DSP00226_V1.1 and Section 7.1 of DSP0227_V1.0

4 Symbols and Abbreviated Terms

4.1

CIM

Common Information Model

4.2

iDRAC

integrated Dell Remote Access Controller – management controller for blades and monolithic servers

4.3

CMC

Chassis Manager Controller – management controller for the modular chassis

4.4

iSCSI

Internet Small Computer System Interface, an Internet Protocol (IP)-based storage networking standard for linking data storage facilities.

4.5

WBEM

Web-Based Enterprise Management

5 Synopsis

Profile Name: BIOS and Boot Management

Version: 1.0.0

Organization: Dell Inc.

CIM Schema Version: 2.19.1

Central Class: DCIM_BIOSService

Scoping Class: CIM_ComputerSystem

The BIOS and Boot Management Profile extends the management capability of the referencing profiles by adding the capability to describe BIOS attributes, each BIOS configuration item is represented by an instance one of these classes DCIM_BIOSEnumeration, DCIMBIOSString, DCIM_BIOSInteger and boot management where each boot list is represented by DCIM_BootConfigSetting and each boot source device by DCIM_BootSourceSetting. DCIM_BIOSService shall be the Central Class.

CIM_ComputerSystem shall be the Scoping Class. The instance of DCIM_BIOSService shall be the Central Instance. The instance of CIM_ComputerSystem with which the Central Instance is associated through the CIM_HostedService association shall be the Scoping Instance.

Table 1 identifies profiles that are related to this profile.

Table 1 – Related Profiles

Profile Name	Organization	Version	Relationship
Profile Registration Profile	DMTF	1.0	Mandatory

6 Description

The BIOS and Boot Management Profile describes BIOS setup configuration including boot management. The profile also describes the relationship of the BIOS classes to the DMTF/Dell profile version information.

Figure 1 represents the class schema for the BIOS attribute management feature of BIOS and Boot Management Profile. For simplicity, the prefix CIM_ has been removed from the names of the classes.

Each BIOS's configurable attribute is represented by one of the classes (DCIM_BIOSEnumeration, DCIM_BIOSString, DCIM_BIOSInteger)CIM_BIOSAttribute. Depending on the dataype of the attribute the BIOS configuration attribute is either instantiaed as DCIM_BIOSEnumeration, DCIM_BIOSString, or DCIM_BIOSInteger instance.

The DCIM_BIOSService class is used to configure the BIOS attributes. The SetAttribute() and SetAttributes() methods on the DCIM_BIOSService class configure BIOS attributes, DCIM_BIOSAttribute subclass instances.

The BIOS and Boot Management Profile information is represented with the instance of CIM_RegisteredProfile.

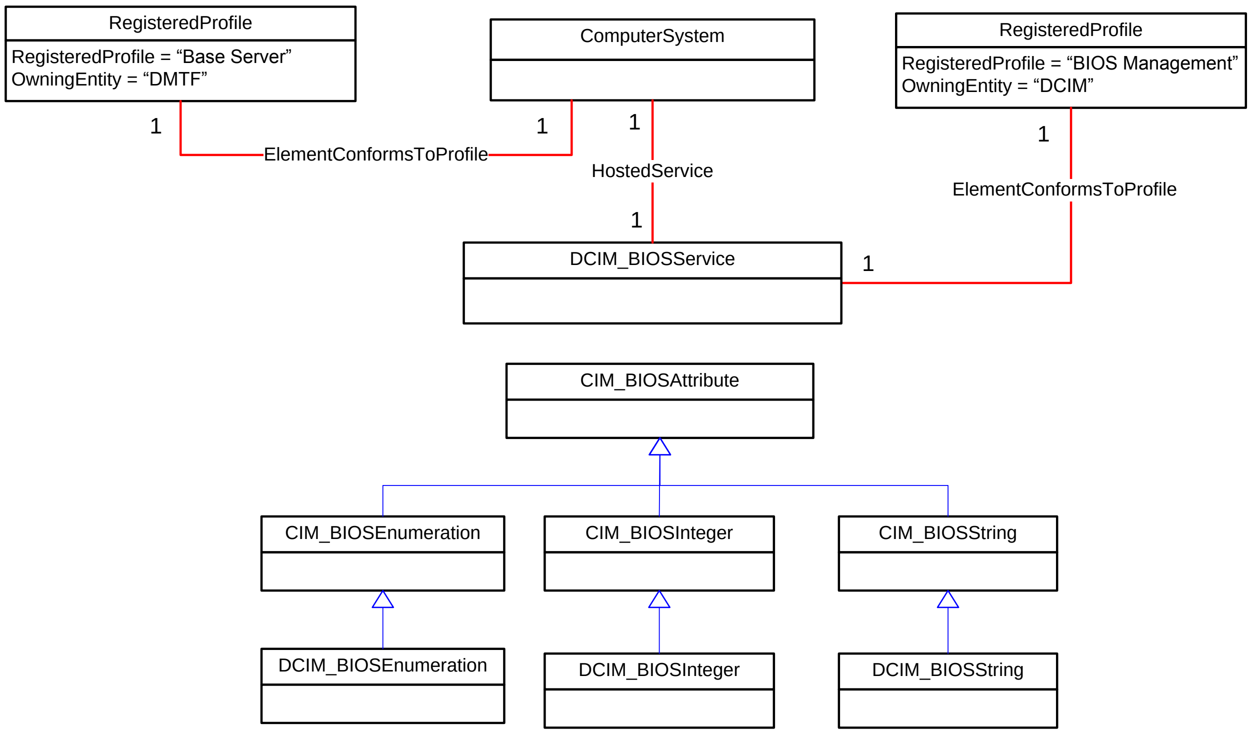

Figure 1 – BIOS and Boot Management Profile: Class Diagram – BIOS Management

Figure 2 represents the class schema for the the boot management feature of the BIOS and Boot Management Profile. For simplicity, the prefix CIM_ has been removed from the names of the classes.

Each boot list is represented by a DCIM_BootConfigSetting instance. Each of the boot lists contains the boot devices/sources represented by DCIM_BootSourceSetting, shown underneath their corresponding boot lists in the figure below. Note that the InstanceID property value prefix of the DCIM_BootSourceSetting instance matches the InstanceID of the DCIM_BootConfigSetting. Also note that IPL boot list contains a BCV boot list.

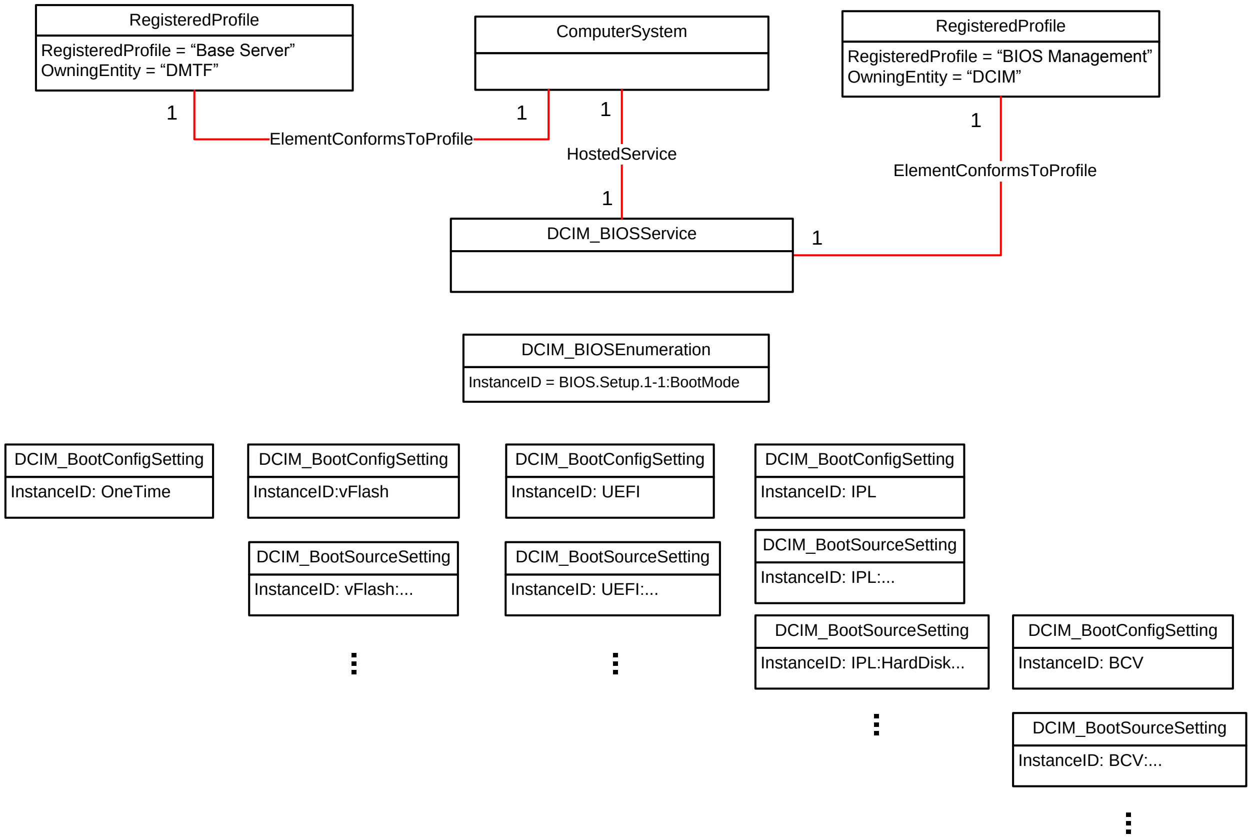

Figure 2 – BIOS and Boot Management Profile: Boot Management

7 Implementation Requirements

Requirements and guidelines for propagating and formulating certain properties of the classes are discussed in this section. Methods are listed in section 8. Table 2 shows the instances of CIM Elements for this profile. Instances of the CIM Elements shall be implemented as described in Table 2. Sections 7 (“Implementation Requirements” and “Methods”) may impose additional requirements on these elements.

Table 2 – CIM Elements: BIOS and Boot Management Profile

Element Name	Requirement	Description
Classes		
DCIM_BIOSService	Mandatory	The class shall be implemented in the Implementation Namespace. See sections 7.3.1
DCIM_BIOSEnumeration	Mandatory	The class shall be implemented in the Implementation Namespace. See section 7.1.1
DCIM_BIOSInteger	Mandatory	The class shall be implemented in the Implementation Namespace. See section 7.1.3
DCIM_BIOSString	Mandatory	The class shall be implemented in the Implementation Namespace. See section 7.1.2
DCIM_BootConfigSetting	Mandatory	The class shall be implemented in the Implementation Namespace. See section **Error! Reference source not found.**
DCIM_BootSourceSetting	Mandatory	The class shall be implemented in the Implementation Namespace. See section **Error! Reference source not found.**
DCIM_LCElementConformsToProfile	Mandatory	The class shall be implemented in the Implementation Namespace.
DCIM_LCElementConformsToProfile	Mandatory	The class shall be implemented in the Interop Namespace.
DCIM_LCRegisteredProfile	Mandatory	The class shall be implemented in the Interop Namespace. See section 7.4.1
Indications		
None defined in this profile		

7.1 BIOS Management

7.1.1 DCIM_BIOSEnumeration

This section describes the implementation for the DCIM_BIOSEnumeration class that represents an enumeration type BIOS attribute..

This class shall be instantiated in the Implementation Namespace.

7.1.1.1 WBEM URIs for WinRM®

The class WBEM URI shall be "http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSEnumeration?__cimnamespace=<Implementation Namespace>"

The key property shall be the InstanceID.

The instance WBEM URI for DCIM_BIOSEnumeration instance shall be: "http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSEnumeration?__cimnamespace=<Implementation Namespace>+InstanceID= BIOS.Setup.1-1:<AttributeName>"

7.1.1.2 Operations

The following table details the implemented operations on DCIM_BIOSEnumeration.

Table 3 – DCIM_BIOSEnumeration - Operations

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

7.1.1.3 Properties

The following table details the implemented properties for DCIM_BIOSEnumeration instance representing a BIOS controller enumeration attribute. The "Requirements" column shall denote the implementation requirement for the corresponding property. If the column "Property Name" matches the property name, the property either shall have the value denoted in the corresponding column "Additional Requirement", or shall be implemented according to the requirements in the corresponding column "Additional Requirement".

Table 4 – Class: DCIM_BIOSEnumeration

Properties	Notes	Additional Requirements
InstanceID	Mandatory	The property value shall be formed as follows: “BIOS.Setup.1-1:<AttributeName property value>”.
AttributeName	Mandatory	The property value shall be from the “AttributeName” column in Table 5
CurrentValue	Mandatory	The property value shall be one of the values in the “PossibleValues” column at the corresponding row in Table 5.
PendingValue	Mandatory	The property value shall be one of the values in the “PossibleValues” column at the corresponding row in Table 5.
IsReadOnly	Mandatory	The property value shall be the value in the “IsReadOnly” column at the corresponding row in Table 5.
FQDD	Mandatory	The property shall be set to “BIOS.Setup.1-1”.
PossibleValues	Mandatory	The property value shall be equal to the array of the values in “PossibleValues” column at the corresponding row in Table 5.

The following table describes the requirements for the AttributeName, and PossibleValues properties. The PossibleValues is an array property represented in the table as comma delimited list.

NOTE: The BIOS attributes listed below may not be present on all Dell systems. The presence of a particular attribute depends on the model of a Dell system, the features available in the system and the BIOS version of the system.

Table 5 – DCIM_BIOSEnumeration Attributes

AttributeName	Attribute Description	IsReadOnly	PossibleValues
MemTest	When set to enabled, the memory tests are performed	FALSE	Enabled, Disabled
RedundantMem		Feature Dependant[1]	Disabled, Spare, Mirror, IntraNodeMirror, DimmSpare
MemOpMode	Memory operating mode	Feature Dependant[1]	OptimizerMode, AdvEccMode, SpareMode, MirrorMode
MemOptimizer	Memory Optimizer Technology	Feature Dependant[1]	Enabled, Disabled
SnoopFilter	Snoop Filter	FALSE	Enabled, Disabled
NodeInterleave	If the system is configured with matching memory, this field enables node interleaving. If set to Disabled (the default), the system supports Non-Uniform Memory architecture (NUMA) (asymmetric) memory configurations. This field is active only if the memory configuration can support node interleaving.	Feature Dependant[1]	Enabled, Disabled

MemVolt	System Memory Voltage	FALSE	AutoVolt, Volt15V, Volt13V
MemLowPower	Memory Low Power Mode	FALSE	Enabled, Disabled
LogicalProc	Each processor core supports up to two logical processors. When this field is set to Enabled (the default), the BIOS reports all logical processors. When set to Disabled, the BIOS only reports one logical processor per core. NOTE: This attribute does not affect the DCIM_CPUView. NumberOfEnabledThreads property because the latter represents the total number of hardware (not BIOS) enabled threads.	FALSE	Enabled, Disabled
ProcVirtualization	Virtualization Technology, When enabled, the additional hardware capabilities provided by Virtualization Technology are available for use	FALSE	Enabled, Disabled
DmaVirtualization	I/O Virtualization Technology	Attribute Value Dependant[2]	Enabled, Disabled
ProcAdjCacheLine	Adjacent Cache Line Prefetch, if Enabled, the system is optimized for applications that require high utilization of sequential memory access.	FALSE	Enabled, Disabled
ProcHwPrefetcher	Hardware Prefetcher, this field enables or disables the hardware prefetcher	FALSE	Enabled, Disabled
DcuStreamerPrefetcher	DCU Streamer Prefetcher	FALSE	Enabled, Disabled
DataReuse	Data Reuse	FALSE	Enabled, Disabled
QpiBandwidthPriority	Intel(R) QPI Bandwidth Priority	FALSE	Compute, InputOutput
ProcExecuteDisable	This field specifies whether Execute Disable Memory Protection Technology is enabled	FALSE	Enabled, Disabled
ProcCores	Number of Cores per Processor	FALSE	All, Single, Dual, Quad, 1, 2, 4, 6, 8, 10, 12 NOTE: For quad port processors, setting attribute ProcCores value to 4 sets the current value to "All".
ProcHyperTransport	HyperTransport Technology	FALSE	HT3, HT1
ProcHtAssist	HT Assist	FALSE	Enabled, Disabled
ProcDramPrefetcher	DRAM Prefetcher	FALSE	Enabled, Disabled
ProcSofwarePrefetcher	Hardware Prefetch	FALSE	Enabled, Disabled

	Training on Software Prefetch		
ProcTurboMode	When set to Enabled, the processor can operate in Turbo Mode	FALSE	Enabled, Disabled
ProcC1E	C1E	FALSE	Enabled, Disabled
ProcCStates	When set to Enabled, the processor can operate in all available Power States	FALSE	Enabled, Disabled
EmbSata	Allows the Embedded SATA to be set to Off or ATA Mode	FALSE	AtaMode, Off
SataPortA	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortB	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortC	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortD	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortE	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortF	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortG	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
SataPortH	Select Auto to enable BIOS support for the device	FALSE	Off, Auto
BootMode	This field determines the boot mode of the system. Selecting 'UEFI' enables booting to Unified Extensible Firmware Interface (UEFI) capable operating systems. Selecting 'BIOS' (the default) ensures compatibility with operating systems that do not support UEFI	FALSE	Bios, Uefi
BootSeqRetry	Boot Sequence Retry, when set to Enabled, the system will re-attempt the Boot Sequence after a 30-second timeout if the last boot attempt has failed.	FALSE	Enabled, Disabled
IntegratedSas	Integrated SAS Controller	FALSE	Enabled, Disabled
IntegratedRaid	Integrated RAID Controller	FALSE	Enabled, Disabled

UsbPorts	User Accessible USB Ports	FALSE	AllOn, OnlyBackPortsOn, AllOff
InternalUsb	Internal USB Port	FALSE	On, Off
InternalUsb1	Internal USB Port	FALSE	On, Off
InternalUsb2	Internal USB Port	FALSE	On, Off
InternalSdCard	Internal SD Card Port	Feature Dependant[1]	On, Off
InternalSdCardRedundancy	Internal SD Card Redundancy	Attribute Value Dependant[3]	Mirror, Disabled
EmbNic1Nic2	Embedded NIC1 and NIC2	FALSE	Enabled, DisabledOs
EmbNic1	Embedded Gb NIC1	FALSE	Enabled, EnabledPxe, EnablediScsi, Disabled
EmbNic2	Embedded Gb NIC2	FALSE	Enabled, EnabledPxe, EnablediScsi, Disabled
EmbNic3Nic4	Embedded NIC3 and NIC4	FALSE	Enabled, DisabledOs
EmbNic3	Embedded Gb NIC3	FALSE	Enabled, EnabledPxe, EnablediScsi, Disabled
EmbNic4	Embedded Gb NIC4	FALSE	Enabled, EnabledPxe, EnablediScsi, Disabled
OsWatchdogTimer	OS Watchdog Timer, if your system stops responding, this watchdog timer aids in the recovery of your operating system. When this field is set to Enabled, the operating system is allowed to initialize the timer. When it is set to Disabled (the default), the timer will have no effect on the system.	FALSE	Enabled, Disabled
IoatEngine	I/OAT DMA Engine, this field enables/disables the I/O Acceleration Technology (I/OAT) option	FALSE	Enabled, Disabled
EmbVideo	Embedded Video Controller	Feature Dependant[1]	Enabled, Disabled
SriovGlobalEnable	SR-IOV Global Enable, this field enables or disables BIOS configuration of Single Root I/O Virtualization (SR-IOV) devices. This feature is disabled by default. Enable if booting to a Virtualization O/S that recognizes SR-IOV devices	FALSE	Enabled, Disabled
SerialComm	Serial Communication	FALSE	OnNoConRedir, OnConRedirCom1, OnConRedirCom2, Off
SerialPortAddress	Serial Port Address	FALSE	Serial1Com1Serial2Com2, Serial1Com2Serial2Com1
ExtSerialConnector	External Serial Connector	FALSE	Serial1, Serial2, RemoteAccDevice

FailSafeBaud	Failsafe Baud Rate	FALSE	115200, 57600, 19200, 9600
ConTermType	Remote Terminal Type	FALSE	Vt100Vt220, Ansi
RedirAfterBoot	Redirection After Boot	FALSE	Enabled, Disabled
FrontLcd	Front-Panel LCD Options	FALSE	None, UserDefined, ModelNum, Advanced
PowerMgmt	Power Management	FALSE	OsCtrl, ActivePwrCtrl, Custom, MaxPerf
ProcPwrPerf	CPU Power and Performance Management	Attribute Value Dependant[4,5]	SysDbpm, MaxPerf, MinPwr, OsDbpm
FanPwrPerf	Fan Power and Performance Management	Attribute Value Dependant[4,6]	MinPwr, MaxPerf
MemPwrPerf	Memory Power and Performance Management	Attribute Value Dependant[4,7]	MaxPerf, 1333MHz, 1067MHz, 978MHz, 800MHz, MinPwr
PasswordStatus	Password Status	FALSE	Unlocked, Locked
TpmSecurity	TPM Security, this field controls the reporting of the Trusted Platform Module (TPM) in the system. When set to Off (default), presence of the TPM is not reported to the OS. When set to On with Pre-boot Measurements, BIOS will store TCG compliant measurements to the TPM during POST. When set to On without Pre-boot Measurements, BIOS will bypass pre-boot measurements.	FALSE	Off, OnPbm, OnNoPbm
TpmActivation	TPM Activation	Attribute Value Dependant[8,9]	NoChange, Activate, Deactivate
TpmClear	Clearing the TPM will cause loss of all keys in the TPM. This could affect booting to OS. When set to Yes, all the contents of the TPM will be cleared.	Attribute Value Dependant[8,10]	No, Yes
TcmSecurity	TCM Security, this field controls the reporting of the Trusted Cryptography Module (TCM) in the system	FALSE	Off, On
TcmActivation	TCM Activation	Attribute Value Dependant[11,12]	NoChange, Activate, Deactivate
TcmClear	Clearing the TCM will cause loss of all keys in the TCM. This could affect booting to OS. When set to Yes, all the contents of the TCM will be cleared. This field is Read-Only when TCM Security is set to Off.	Attribute Value Dependant[11,13]	Yes, No
PwrButton	This field enables/disables the power button on the	FALSE	Enabled, Disabled

	front panel.		
NmiButton	This field enables/disables the NMI button on the front panel.	FALSE	Enabled, Disabled
AcPwrRcvry	AC Power Recovery, this field specifies how the system will react after AC power has been restored to the system. It is especially useful for people who turn their systems off with a power strip. When set to Off, the system will stay off after AC is restored. When set to On, the system will turn on after AC is restored. When set to Last, the system will turn on if the system was on when AC was lost. The system will remain off if the system was off when AC was lost	FALSE	Last, On, Off
AcPwrRcvryDelay	AC Power Recovery Delay, this field specifies how the system will support the staggering of power-up after AC power has been restored to the system. When set to Immediate, there is no delay for power-up. When set to Random, the system will create a random delay (30s to 240s) for power-up. When set to User Defined, the system will delay power-up by that amount. The system supported user defined power-up delay range is from 30s to 240s.	FALSE	Immediate, Random, User
NumLock	Keyboard NumLock.	FALSE	On, Off
ReportKbdErr	Report Keyboard Errors.	FALSE	Report, NoReport
ErrPrompt	F1/F2 Prompt on Error.	FALSE	Enabled, Disabled

NOTE: 1 – The attribute's read-only status (IsReadOnly property value) depends on the particular platform model, or platform features, or the platform's bios version.

NOTE: 2 – The DmaVirtualization is read-only (IsReadOnly=TRUE) and shall have value "Disabled", if the ProcVirtualization attribute is set to "Disabled". InternalSdCard

NOTE: 3 – The InternalSdCardRedundancy is read-only (IsReadOnly=TRUE) and shall have value "Disabled", if the InternalSdCard attribute is set to "Disabled".

NOTE: 4 – The ProcPwrPerf, FanPwrPerf, MemPwrPerf attributes are settable (IsReadOnly=FALSE), if the PowerMgmt attribute is set to "Custom"; otherwise those attributes are read-only (IsReadOnly=TRUE).

NOTE: 5 – The ProcPwrPerf shall have value:

- "OsDbpm" if PowerMgmt is set to "OsCtrl".
- "SysDbpm" if PowerMgmt is set to "ActivePwrCtrl"

- “MaxPerf” if PowerMgmt is set to “MaxPerf”

NOTE: 6 – The FanPwrPerf shall have value “MinPwr” if PowerMgmt is set to “OsCtrl” or “ActivePwrCtrl”.

NOTE: 7 – The MemPwrPerf shall have value “MinPwr” if PowerMgmt is not set to “Custom”.

NOTE: 8 – The TpmActivation, TpmClear attributes are settable (IsReadOnly=FALSE), if the TpmSecurity attribute is NOT set to “Off”; otherwise those attributes are read-only (IsReadOnly=TRUE).

NOTE: 9 – The TpmActivation shall have value “NoChange”, if TpmSecurity is set to “Off”.

NOTE: 10 – The TpmClear shall have value “No”, if TpmSecurity is set to “Off”.

NOTE: 11 – The TcmActivation, TcmClear attributes are settable (IsReadOnly=FALSE), if the PowerMgmt attribute is set to “Custom”; otherwise those attributes are read-only (IsReadOnly=TRUE).

NOTE: 12 – The TcmActivation shall have value “NoChange”, if TcmSecurity is set to “Off”.

NOTE: 13 – The TcmClear shall have value “No”, if TcmSecurity is set to “Off”.

7.1.2 DCIM_BIOSString

This section describes the implementation for the DCIM_BIOSString class that represents a string type BIOS attribute.

This class shall be instantiated in the Implementation Namespace.

7.1.2.1 WBEM URIs for WinRM®

The class WBEM URI shall be “http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSString?__cimnamespace=<Implementation Namespace>”

The key property shall be the InstanceID.

The instance WBEM URI for DCIM_BIOSString instance shall be:
http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSString?__cimnamespace=<Implementation Namespace>+InstanceID=BIOS.Setup.1-1:<AttributeName> (AttributeName comes from Table 5)

7.1.2.2 Operations

The following table details the implemented operations on DCIM_BIOSString.

Table 6 – DCIM_BIOSString - Operations

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

7.1.2.3 Properties

The following table details the implemented properties for DCIM_BIOSString instance representing a BIOS string attribute. The “Requirements” column shall denote the implementation requirement for the corresponding property. If the column “Property Name” matches the property name, the property either shall have the value denoted in the corresponding column “Additional Requirement”, or shall be implemented according to the requirements in the corresponding column “Additional Requirement”.

Table 7 – Class: DCIM_BIOSString

Properties	Notes	Additional Requirements
InstanceID	Mandatory	The property value shall be formed as follows: BIOS.Setup.1-1:<AttributeName property value>".
AttributeName	Mandatory	The property value shall be from the "AttributeName" column in Table 8.
CurrentValue	Mandatory	The property value shall match the format described in "Value Expression" column at the corresponding row in Table 8.
PendingValue	Mandatory	The property value shall match the format described in "Value Expression" column at the corresponding row in Table 8.
IsReadOnly	Mandatory	The property value shall be the value in the "IsReadOnly" column at the corresponding row in Table 8.
FQDD	Mandatory	The property shall be set to "BIOS.Setup.1-1".
MinLength	Mandatory	The property value shall be the value in the "MinLength" column at the corresponding row in Table 8.
MaxLength	Mandatory	The property value shall be the value in the "MaxLength" column at the corresponding row in Table 8.

The following table describes possible DCIM_BIOSString attributes and the requirements for the AttributeName, MinLength, and MaxLength properties.

NOTE: The BIOS attributes listed below may not be present on all Dell systems. The presence of a particular attribute depends on the model of a Dell system, the features available in the system and the BIOS version of the system.

Table 8 – DCIM_BIOSString Attributes

AttributeName	Attribute Description	IsReadOnly	MinLength	MaxLength	Value Expression
AssetTag	AssetTag	FALSE	0	10	String
UserLcdStr	User-Defined LCD String	FALSE			String

7.1.3 DCIM_BIOSInteger

This section describes the implementation for the DCIM_BIOSInteger class that represents an integer type BIOS attribute.

This class shall be instantiated in the Implementation Namespace.

7.1.3.1 WBEM URIs for WinRM®

The class WBEM URI shall be "http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSInteger?__cimnamespace=<Implementation Namespace>"

The key property shall be the InstanceID.

The instance WBEM URI for DCIM_BIOSInteger instance shall be: "http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSInteger?__cimnamespace=<Implementation Namespace>+InstanceID= BIOS.Setup.1-1:AttributeName (AttributeName comes from Table 5)"

7.1.3.2 Operations

The following table details the implemented operations on DCIM_BIOSInteger.

Table 9 – DCIM_BIOSInteger - Operations

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

7.1.3.3 Properties

The following table details the implemented properties for DCIM_BIOSInteger instance representing a BIOS integer attribute. The "Requirements" column shall denote the implementation requirement for the corresponding property. If the column "Property Name" matches the property name, the property either shall have the value denoted in the corresponding column "Additional Requirement", or shall be implemented according to the requirements in the corresponding column "Additional Requirement".

Table 10 – Class: DCIM_BIOSInteger

Properties	Notes	Additional Requirements
InstanceID	Mandatory	The property value shall be formed as follows: "BIOS.Setup.1-1:<AttributeName property value>".
AttributeName	Mandatory	The property value shall be from the "AttributeName" column in Table 811.
CurrentValue	Mandatory	The property value shall match the format described in "Value Expression" column at the corresponding row in Table 8.
PendingValue	Mandatory	The property value shall match the format described in "Value Expression" column at the corresponding row in Table 8.
IsReadOnly	Mandatory	The property value shall be the value in the "IsReadOnly" column at the corresponding row in Table 8.
FQDD	Mandatory	The property shall be set to "BIOS.Setup.1-1".
LowerBound	Mandatory	The property value shall be the value in the "LowerBound" column at the corresponding row in Table 8.
UpperBound	Mandatory	The property value shall be the value in the "UpperBound" column at the corresponding row in Table 8.

The following table describes possible DCIM_BIOSInteger attributes and the requirements for the AttributeName, IsReadOnly, LowerBound, and UpperBound properties.

NOTE: The BIOS attributes listed below may not be present on all Dell systems. The presence of a particular attribute depends on the model of a Dell system, the features available in the system and the BIOS version of the system.

Table 11 – DCIM_BIOSInteger Attributes

AttributeName	Attribute Description	IsReadOnly	LowerBound	UpperBound
AcPwrRcvryUserDelay	AC Power Recovery User Defined Delay	TRUE	30	240

7.2 Boot Management

Each of DCIM_BootConfigSetting instances shall represent a boot list, and each boot list can be enabled to be used in the next boot using the algorithm in “Boot State Enablement” column. The following boot lists shall be implemented:

Table 12 – Boot Lists

Boot Lists	DCIM_BootConfigSetting.InstanceID	Boot State Enablement	Description
IPL/BIOS	IPL	SetAttribute() or SetAttributes() method with AttributeName “BootMode” and AttributeValue “Bios”	IPL list of boot devices as defined in the BIOS Boot Specification. IPL list represents the traditional BIOS boot list.
BCV	BCV	SetAttribute() or SetAttributes() method with AttributeName “BootMode” and AttributeValue “Bios”	BCV list of boot devices as defined in the BIOS Boot Specification. BCV list usually contains the list of storage controllers for booting from a particular hard drive. NOTE: BCV list is a nested list within the IPL list. Selecting “Hard drive C” in the IPL boot list selects the BCV list for booting.
UEFI	UEFI	SetAttribute() or SetAttributes() method with AttributeName “BootMode” and AttributeValue “Uefi”	List of UEFI devices for boot.
vFlash Partition	vFlash	ChangeBootOrderByInstanceID() on DCIM_BootConfigSetting with InstanceID “OneTime” and source[] containing a single vFlash DCIM_BootSourSetting InstanceID	vFlash partitions that could be booted from.
One Time Boot	OneTime	ChangeBootOrderByInstanceID() on DCIM_BootConfigSetting with InstanceID “OneTime” and source[] containing a single DCIM_BootSourSetting InstanceID from any boot list.	One time boot list contains a single boot device selected for one time boot. After the reboot, the boot list reverts to the original boot list.

The DCIM_BootSourceSetting.InstanceID value shall represent the owning boot list instance of DCIM_BootConfigSetting, where the prefix substring value before the first colon shall match the DCIM_BootConfigSetting.InstanceID value.

> For example: DCIM_BootSourceSetting.InstanceID with value of “**vFlash**:LABEL1:1” belongs to DCIM_BootConfigSetting boot list with InstanceID “**vFlash**”.

The BCV boot device that belongs to the IPL list and represents the BCV list, shall have DCIM_BootSourceSetting.InstanceID property value with prefix substring: “IPL:HardDisk”.

All the boot devices within the list may be sorted using the ChangeBootOrderByInstanceID() method (section 8.7) and may be enabled/disabled using the ChangeBootSourceState() method (section 8.6).

The DCIM_BootConfigSetting.IsCurrent, IsNext and IsDefault properties shall represent the current state of the boot list.

- The IsNext property set to 1(Is Next) shall represents that the boot list is configured to be used for the next boot. vFlash boot list shall not have this value.
- The IsNext property set to 3(Is Next for Single Use) shall represent that the boot list is configured to be used ONLY for the next boot. Only the OneTime boot list may have this value for the IsNext property.

The state of the boot list for the next boot shall be changed through the DCIM_BIOSEnumeration with AttributeName “BootMode” (section Table 5) or through execution of ChangeBootOrderByInstanceID() method on the DCIM_BootConfigSetting instance with InstanceID “OneTime” with the source[] parameter having a single DCIM_BootSourceSetting InstanceID from any of the lists including vFlash.

Each boot list contains boot devices that shall be represented by DCIM_BootSourceSetting.

7.2.1 DCIM_BootConfigSetting

This section describes the implementation for the DCIM_BootConfigSetting class that represents a particular boot list.

This class shall be instantiated in the Implementation Namespace.

7.2.1.1 WBEM URIs for WinRM®

The class WBEM URI shall be “http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BootConfigSetting?__cimnamespace=<Implementation Namespace>”

The key property shall be the InstanceID.

The instance WBEM URI for DCIM_BootConfigSetting instance shall be: “http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BootConfigSetting?__cimnamespace=<Implementation Namespace>+InstanceID=<a value from Table 12 DCIM_BootConfigSetting.InstanceID column>”

7.2.1.2 Operations

The following table details the implemented operations on DCIM_BootConfigSetting.

Table 13 – DCIM_BootConfigSetting – Operations

Operation Name	Requirements	Required Input
Get	Mandatory	Instance URI
Enumerate	Mandatory	Class URI
Invoke	Mandatory	Instance URI
DCIM_BIOSService.SetAttribute()	Mandatory	See section 8.1 with AttributeName = “BootMode”
DCIM_BIOSService.SetAttributes()	Mandatory	See section 8.2 with AttributeName = “BootMode”

7.2.1.3 Properties

The following table details the implemented properties for DCIM_BootConfigSetting. The “Requirements” column shall denote the implementation requirement for the corresponding property. If the column “Property Name” matches the property name, the property either shall have the value denoted in the corresponding column “Additional Requirement”, or shall be implemented according to the requirements in the corresponding column “Additional Requirement”.

Table 14 – Class: DCIM_BootConfigSetting

Properties and Methods	Requirement	Description
InstanceID	Mandatory	The property value shall be from Table 12 “DCIM_BootConfigSetting.InstanceID” column.
ElementName	Mandatory	
IsCurrent	Mandatory	Value of 1 = Is Current (Is the current boot configuration), Value of 2 = Is Not Current (Is not the current boot configuration)
IsDefault	Mandatory	Value of 1 = Is Default (is the default boot configuration) Value of 2 = Is Not Default (is not the default boot configuration)
IsNext	Mandatory	Value of 1 = Is Next (is the next boot configuration the system will use for booting) Value of 2 = Is Not Next (is not the next boot configuration the system will use for booting) Value of 3= Is Next For Single Use (is the next boot configuration the system will use for booting for single use, one time boot only)

7.2.2 DCIM_BootSourceSetting

This section describes the implementation for the DCIM_BootSourceSetting class that represents a boot device.

This class shall be instantiated in the Implementation Namespace.

7.2.2.1 WBEM URIs for WinRM®

The class WBEM URI shall be “http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BootSourceSetting?__cimnamespace=<Implementation Namespace>”

The key property shall be the InstanceID.

The instance WBEM URI for DCIM_BootSourceSetting instance shall be: “http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BootSourceSetting?__cimnamespace=<Implementation Namespace>+InstanceID=<InstanceID see Table 16>”

7.2.2.2 Operations

The following table details the implemented operations on DCIM_BootSourceSetting.

Table 15 – DCIM_BootSourceSetting – Operations

Operation Name	Requirements	Required Input
Get	Mandatory	Instance URI
Enumerate	Mandatory	Class URI
DCIM_BootConfigSetting.ChangeBootSourceState()	Mandatory	See section 8.6.
DCIM_BootConfigSetting.ChangeBootOrderByInstanceID	Mandatory	See section 8.7

7.2.2.3 Properties

The following table details the implemented properties for DCIM_BootSourceSetting. The “Requirements” column shall denote the implementation requirement for the corresponding property. If the column “Property Name” matches the property name, the property either shall have the value denoted in the corresponding column “Additional Requirement”, or shall be implemented according to the requirements in the corresponding column “Additional Requirement”.

Table 16 – Class: DCIM_BootSourceSetting

Properties and Methods	Requirement	Description
InstanceID	Mandatory	The property value shall have prefix from Table 12 “DCIM_BootSourceSetting.InstanceID” column followed by a unique id representing the boot source. For example: UEFI:Disk.USBFront.2-1:3156051d1529b8f4f88c99f54b895350 (boot source belongs to **UEFI** bootlist) IPL:NIC.Slot.4-2:d0f2c6c736adb8c2238153293a0c026c (boot source belongs to **IPL** bootlist) BCV:RAID.Integrated.1-1:b84a10539d2ccaca5e86b7de3cae08a8 (boot source belongs to **BCV** bootlist)
BIOSBootString	Mandatory	Descriptive boot source name
BootString	Mandatory	Descriptive boot source name
PendingAssignedSequence	Mandatory	The value shall be set through the successful execution of the ChangeBootOrderByInstanceID() method, indicates the Pending Assigned Sequence of this instance.
CurrentAssignedSequence	Mandatory	The *CurrentAssignedSequence* attribute of this instance defines the its place in the zero based indexed boot sequence.
PendingEnabledStatus	Mandatory	The value shall be set through the successful execution of the ChangeBootSourceState () method, indicates the Pending Enabled Status of this instance.
CurrentEnabledStatus	Mandatory	The *CurrentEnabledStatus* attribute of this instance identifies whether it is enabled or disabled, if disabled this boot source will not be attempted for boot while booting from the bootlist.
ElementName	Mandatory	
FailThroughSupported	Mandatory	An enumeration indicating the behavior when the attempt to boot using the boot source fails (no media, timeout). The current values in the enumeration are: 0 = Unknown 1 = Is Supported 2 = Is Not Supported A value of 1 (Is Supported) indicates that next boot source the boot order is used. A value of 2 (Is Not Supported) indicates that the boot order is terminated and no other boot sources are used.

7.3 Service for Method Invocations

7.3.1 DCIM_BIOSService

This section describes the implementation for the DCIM_BIOSService class that represents the BIOS and boot management service.

This class shall be instantiated in the Implementation Namespace.

The DCIM_LCElementConformsToProfile association(s) shall reference the DCIM_BIOSService instance(s).

7.3.1.1 WBEM URIs for WinRM®

The class WBEM URI shall be "http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSService?__cimnamespace=<Implementation Namespace>"

The key properties shall be SystemCreationClassName, CreationClassName, SystemName and Name.

The instance WBEM URI for DCIM_BIOSService instance shall be: "http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSService?__cimnamespace=<Implementation Namespace>+SystemCreationClassName=DCIM_ComputerSystem+CreationClassName=DCIM_BIOSService+SystemName=DCIM:ComputerSystem+Name=DCIM:BIOSService"

7.3.1.2 Operations

The following table details the implemented operations on DCIM_BIOSService.

Table 17 – DCIM_BIOSService – Operations

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

7.3.1.3 Properties

The following table details the implemented properties for DCIM_BIOSService instance representing a system in a system. The "Requirements" column shall denote the implementation requirement for the corresponding property. If the column "Property Name" matches the property name, the property either shall have the value denoted in the corresponding column "Additional Requirement", or shall be implemented according to the requirements in the corresponding column "Additional Requirement".

Table 18 – Class: DCIM_BIOSService

Properties and Methods	Requirement	Description
SystemCreationClassName	Mandatory	The property value shall be “DCIM_ComputerSystem”.
CreationClassName	Mandatory	The property value shall be “DCIM_BIOSService”.
SystemName	Mandatory	The property value shall be “DCIM:ComputerSystem”.
Name	Mandatory	The property value shall be “DCIM:BIOSService”
ElementName	Mandatory	The property value shall be “BIOS Service”

7.4 Profile Registration

7.4.1 BIOS and Boot Management Profile Registration

This section describes the implementation for the DCIM_LCRegisteredProfile class.

This class shall be instantiated in the Interop Namespace.

The DCIM_ElementConformsToProfile association(s) shall reference the DCIM_LCRegisteredProfile instance.

7.4.1.1 WBEM URIs for WinRM®

The class WBEM URI shall be "http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_RegisteredProfile?__cimnamespace=<Interop Namespace>"

The key property shall be the InstanceID property.

The instance WBEM URI shall be: “http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_LCRegisteredProfile?__cimnamespace=<InteropNamespace>+InstanceID=DCIM:BIOSandBootManagement:1.0.0”

7.4.1.2 Operations

The following table details the implemented operations on DCIM_LCRegisteredProfile.

Table 19 – DCIM_LCRegisteredProfile - Operations

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

7.4.1.3 Properties

The following table details the implemented properties for DCIM_LCRegisteredProfile instance representing BIOS and Boot Management Profile implementation. The “Requirements” column shall denote the implementation requirement for the corresponding property. If the column “Name” matches the property name, the property either shall have the value denoted in the corresponding column “Additional Requirements”, or shall be implemented according to the requirements in the corresponding column “Additional Requirements”.

Table 20 – Class: CIM_RegisteredProfile

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

8 Methods

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

8.1 CIM_BIOSService.SetAttribute( )

The SetAttribute( ) method is used to set or change the value of a BIOS attribute.

Invocation of the SetAttribute( ) method shall change the value of the attribute’s CurrentValue or attribute’s PendingValue property to the value specified by the AttributeValue parameter if the attribute’s IsReadOnly property is FALSE. Invocation of this method when the attribute’s IsReadOnly property is TRUE shall result in no change to the value of the attribute’s CurrentValue property. The results of changing this value are described with the SetResult parameter.

Return code values for the SetAttribute( ) method are specified in Table 21 and parameters are specified in Table 22.Invoking the SetAttribute( ) method multiple times can result in the earlier requests being overwritten or lost.

Table 21 – SetAttribute( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Failed

Table 22 – SetAttribute( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Target	String	Shall be set to “BIOS.Setup.1-1”
IN, REQ	AttributeName	String	Shall contain the AttributeName property value for the attribute to be modified.
IN, REQ	AttributeValue[]	String	Shall contain the desired attribute value. If the value is valid, the CurrentValue or PendingValue property of the specified attribute will be modified.
OUT	SetResult	String	Returns: "Set CurrentValue property" when the attributes current value is set. “Set PendingValue property” when the attributes pending value is set.
OUT	RebootRequired	String	Returns: “Yes” if reboot is required, “No” if reboot is not required.
OUT	MessageID	String	Error MessageID
OUT	Message	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

8.2 DCIM_BIOSService.SetAttributes( )

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

Invocation of the SetAttributes( ) method shall change the values of the attribute’s CurrentValue or PendingValue properties that correspond to the names specified by the AttributeName parameter and the values specified by the AttributeValue parameter if the respective attribute’s IsReadOnly property is FALSE. Invocation of this method when the respective attribute’s IsReadOnly property is TRUE shall result in no change to the corresponding value of the attribute’s CurrentValue property.

Return code values for the SetAttributes( ) method are specified in Table 23, and parameters are specified in Table 24.

Invoking the SetAttributes( ) method multiple times can result in the earlier requests being overwritten or lost.

Table 23 – SetAttributes( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Failed

Table 24 – SetAttributes( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Target	String	Shall be set to “BIOS.Setup.1-1”
IN, REQ	AttributeName[]	String	Shall contain the AttributeName property value for the attribute to be modified.
IN, REQ	AttributeValue[]	String	Shall contain the desired attribute values. If the value is valid, the CurrentValue or PendingValue property of the specified attribute will be modified.
OUT	SetResult[]	String	Returns: "Set CurrentValue property" when the attributes current value is set. “Set PendingValue property” when the attributes pending value is set.
OUT	RebootRequired[]	String	Returns: “Yes” if reboot is required, “No” if reboot is not required.
OUT	MessageID[]	String	Error MessageID
OUT	Message[]	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

8.3 DCIM_BIOSService.CreateTargetedConfigJob( )

The CreateTargetedConfigJob( ) method is used to apply the pending values created by the SetAttribute, SetAttributes, ChangeBootSourceState and ChangeBootOrderByInstanceID methods. The successful execution of this method creates a job for application of pending values.

CreateTargetedConfigJob method supports the following optional input parameters

1. RebootJobType: when provided in the input parameters, creates a specific reboot job to "PowerCycle" or "Graceful Reboot without forced shutdown" or "Graceful Reboot with forced shutdown". This parameter only creates the RebootJob and does not schedule it.

2. ScheduledStartTime: When provided in the input parameters, schedules the "configuration job" and the optional "reboot job" at the specified start time. A special value of "TIME_NOW" schedules the job(s) immediately.

3. UntilTime: This parameter has a dependency on "ScheduledStartTime", together "ScheduledStartTime" and "UntilTime" define a time window for scheduling the job(s). Once scheduled, jobs will be executed within the time window.

If CreateTargetedConfigJob method is executed without the 3 optional parameters discussed above, then configuration job is created but not scheduled. However, this configuration job can be scheduled later using the DCIM_JobService.SetupJobQueue () method from the "Job Control Profile". DCIM_JobService.SetupJobQueue () can be executed to schedule several configuration jobs including the reboot job. Refer to "Job Control Profile" for more details.

Return code values for the CreateTargetedConfigJob( ) method are specified in Table 23, and parameters are specified in Table 24.

Subsequent calls to CreateTargetedConfigJob after the first CreateTargetedConfigJob will result in error until the first job is completed."

Table 25 – CreateTargetedConfigJob( ) Method: Return Code Values

Value	Description
0	Success
1	Not supported
2	Failed
4096	Job Created

Table 26 – CreateTargetedConfigJob( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Target	String	Shall be set to "BIOS.Setup.1-1"
IN	RebootJobType	Uint16	Shall contain the requested reboot type: 1 - PowerCycle 2 - Graceful Reboot without forced shutdown 3 - Graceful Reboot with forced shutdown.
IN	ScheduledStartTime	String	Start time for the job execution in format: yyyymmddhhmmss. The string "TIME_NOW" means immediate.
IN	UntilTime	String	End time for the job execution in format: yyyymmddhhmmss. : If this parameter is not NULL, then ScheduledStartTime parameter shall also be specified.
OUT	Job	CIM_ConcreteJob REF	Reference to the newly created pending value application job.
OUT	MessageID	String	Error MessageID
OUT	Message	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

8.4 DCIM_BIOSService.DeletePendingConfiguration( )

The DeletePendingConfiguration( ) method is used to cancel the pending values created by the SetAttribute and SetAttributes methods. The DeletePendingConfiguration( ) method cancels the pending configuration changes made before the configuration job is created with CreateTargetedConfigJob(). This method only operates on the pending changes prior to CreateTargetedConfigJob() being called. After the configuration job is created, the pending changes can only be canceled by calling DeleteJobQueue() method in the Job Control profile.

Return code values for the DeletePendingConfiguration( ) method are specified in Table 27, and parameters are specified in Table 28.

Table 27 – DeletePendingConfiguration( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Failed

Table 28 – DeletePendingConfiguration( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Target	String	FQDD of the BIOS
OUT	MessageID	String	Error MessageID
OUT	Message	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

8.5 DCIM_BIOSService.ChangePassword( )

The ChangePassword( ) method is used to set the system and setup password in BIOS setting.

Return code values for the ChangePassword( ) method are specified in , and parameters are specified in Table 34.

For old and new passwords, use “null” or not sending it for no password (case sensitive).

Table 33 – ChangePassword( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Failed

Table 34 – ChangePassword( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Target	String	FQDD of the BIOS

Qualifiers	Name	Type	Description/Values
IN, REQ	PasswordType	Uint16	Password Type. 1 – System Password 2-Setup Password
IN, REQ	OldPassword	string	Old Password null or string
IN, REQ	NewPassword	string	New Password null or string
OUT	MessageID	String	Error MessageID
OUT	Message	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

8.6 DCIM_BootConfigSetting.ChangeBootSourceState( )

The ChangeBootSourceState( ) method is used change the enabled/disabled state of a single or multiple boot devices.

The successful invocation of the ChangeBootSourceState( ) method shall change the boot sources state and affect DCIM_BootSourceSetting.PendingEnabledStatus properties. Upon the successful invocation, the DCIM_BootSourceSetting.PendingEnabledStatus shall have the value specified by the EnabledState parameter for the DCIM_BootSourceSetting instances with the InstanceID property matching the InstanceID parameter value(s).

Upon the successful completion of the returned job, the CurrentEnabledStatus shall have the same value as the PendingEnabledStatus.

Return code values for the ChangeBootSourceState( ) method are specified in Table 29 and parameters are specified in Table 30. Invoking the ChangeBootSourceState( ) method multiple times can result in the earlier requests being overwritten or lost.

Table 29 – ChangeBootSourceState( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Failed
4096	Job Created

Table 30 – ChangeBootSourceState( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	EnabledState	String	Shall contain the requested state for the boot device.
IN, REQ	source[]	String	Shall contain the InstanceID value(s) for DCIM_BootSourceSetting instances to be affected.
OUT	Job	CIM_Concrete Job REF	Reference to the newly created pending value application job.
OUT	MessageID	String	Error MessageID
OUT	Message	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

8.7 DCIM_BootConfigSetting.ChangeBootOrderByInstanceID( )

The ChangeBootOrderByInstanceID( ) method is used to change the order of boot devices within the boot list.

The successful invocation of the ChangeBootOrderByInstanceID( ) method shall order the boot devices in the list in accordance to the corresponding array element in the “source” parameter array. The omitted boot devices in the “source” parameter array shall be omitted in the boot list ordering.

Each element of the “source” parameter array shall have value of a DCIM_BootSourceSetting.InstanceID property.

Upon successful completion of this method, the value of the PendingAssignedSequence property on each instance of CIM_BootSourceSetting shall be updated such that the values are monotonically increasing in correlation with the position of the referenced DCIM_BootSourceSetting instance in the “source” input parameter array. That is, the first position in the array shall have the lowest non-zero value for PendingAssignedSequence. The second position will have the second lowest value, and so on.

Upon successful completion of this method, the value of the PendingAssignedSequence property on each instance of DCIM_BootSourceSetting, that relates to the target DCIM_BootConfigSetting instance that is not present in the input array, shall be assigned a value of 0.

Upon the successful completion of the returned job, the CurrentAssignedSequence shall have the same value as the PendingAssignedSequence.

Return code values for the ChangeBootOrderByInstanceID( ) method are specified in Table 31 and parameters are specified in Table 32. Invoking the ChangeBootOrderByInstanceID( ) method multiple times can result in the earlier requests being overwritten or lost.

Table 31 – ChangeBootOrderByInstanceID( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Failed
4096	Job Created

Table 32 – ChangeBootOrderByInstanceID( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	source[]	String	Shall contain the InstanceID value(s) for DCIM_BootSourceSetting instances to change the order of.
OUT	Job	CIM_Concrete Job REF	Reference to the newly created pending value application job.
OUT	MessageID	String	Error MessageID
OUT	Message	String	Error Message
OUT	MessageArguments[]	String	Error MessageArguments

9 Use Cases

This section contains use cases for the Dell BIOS and Boot Profile.

Note that URIs in this section are in form of WBEM URIs for WinRM®.

9.1 Discovery of BIOS and Boot profile support

Use one of the two procedures below to confirm the existence of BIOS and Boot profile support

A) `GET` the *DCIM_LCRegisteredProfile* instance using an *InstanceID* of DCIM:BIOSandBootManagement:1.0.0. See section 3.14 for a definition of `GET`.

Instance URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_LCRegisteredProfile?__cimnamespace=root/interop+InstanceID=DCIM:BIOSandBootManagement:1.0.0

Results for the *InstanceID* of DCIM:BIOSandBootManagement:1.0.0 shown below. If no instance is returned, the profile is not supported.

DCIM_LCRegisteredProfile
AdvertiseTypeDescriptions = WS-Identify, Interop Namespace
AdvertiseTypes = 1, 1
InstanceID = DCIM:BIOSandBootManagement:1.0.0
OtherRegisteredOrganization = DCIM
RegisteredName = BIOS and Boot Management
RegisteredOrganization = 1
RegisteredVersion = 1.0.0

B) `ENUMERATE` the *CIM_RegisteredProfile* class. See section 3.13 for a definition of `ENUMERATE`.

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_RegisteredProfile?__cimnamespace=root/interop

Then query the result for the following properties:

RegisteredName = BIOS and Boot Management, OtherRegisteredOrganization = DCIM, RegisteredVersion = 1.0.0

9.2 Inventory of BIOS attributes in system

`ENUMERATE` the DCIM_BIOSEnumeration class to view all available instances of the class. For the class and general instance URI structure, see section 7.1.1.1

Class URI:

http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSEnumeration?__cimnamespace=root/dcim

The instance information of all available BIOS attributes will be returned

9.3 Get the first BIOS attribute's information

The URI for getting particular instance information is deterministic (i.e the *InstanceID* will be unique for each instance)

For the first BIOS attribute in the system, the instance URI will be:

http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_BIOSEnumeration?__cimnamespace=root/dcim+InstanceID=BIOS.Setup.1-1:MemTest

The instance of *DCIM_BIOSEnumeration* that contains the information on the first BIOS attribute will be returned

9.4 Setting BIOS attributes

A) `ENUMERATE` the *DCIM_BIOSEnumeration* class as shown in section 9.2 and identify the applicable instances

B) Confirm the *IsReadOnly* field is set to false

C) To invoke the SetAttribute() or SetAttributes() method, extract the instance information from A) and construct the input parameters per Table 24

D) `INVOKE` the SetAttribute() or SetAttributes() method

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_BIOSService?SystemCreationClassName=DCIM_ComputerSystem+CreationClassName=DCIM_BIOSService+SystemName=DCIM:ComputerSystem+Name=DCIM:BIOSService

E) Examine output parameters per Table 23.

F) Apply the pending values (Section 9.5)

G) Repeat A) to confirm successful execution of the method

9.5 Apply pending values

A) To invoke the CreateTargetedConfigJob( ) method, construct input parameters per Table 26 and use the BIOS FQDD from section 9.2

B) `INVOKE` CreateTargetedConfigJob( ) method

Class URI:
http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_BIOSService?SystemCreationClassName=DCIM_ComputerSystem+CreationClassName=DCIM_BIOSService+SystemName=DCIM:ComputerSystem+Name=DCIM:BIOSService

C) Allow several minutes for the UEFI to execute the SSIB task, which will be followed by a reboot of the system

D) Query the status of the *jobID* output using the job control profile methods

9.6 Delete pending values

A) To invoke the DeletePendingConfiguration( ) method, construct input parameters per Table 28 and use the BIOS FQDD from section 9.2

B) `INVOKE` DeletePendingConfiguration( ) method

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_BIOSService?SystemCreationClassName=DCIM_ComputerSystem+CreationClassName=DCIM_BIOSService+SystemName=DCIM:ComputerSystem+Name=DCIM:BIOSService

C) If the return parameters indicate success**,** per Table 27, no further action necessary

9.7 Inventory of boot configurations in system

`ENUMERATE` the *DCIM_BootConfigSetting* class to view all available instances of the class. For the class and general instance URI structure, see section section 7.2.1.1

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BootConfigSetting?__cimnamespace=root/dcim

The instance information of all available boot configurations will be returned

9.8 Get the first boot configuration's information

The URI for getting particular instance information is deterministic (i.e the *InstanceID* will be unique for each instance)

For the first boot configuration in the system, the instance URI will be:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BootConfigSetting?__cimnamespace=root/dcim+InstanceID=IPL

The instance of *DCIM_BootConfigSetting* that contains the information on the first boot configuration will be returned

9.9 Inventory of boot sources in system

`ENUMERATE` the *DCIM_BootSourceSetting* class to view all available instances of the class. For the class and general instance URI structure, see section section 7.2.2.1

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BootSourceSetting?__cimnamespace=root/dcim

The instance information of all available boot sources will be returned

9.10 Changing boot order by instance

A) `ENUMERATE` the *DCIM_BootConfigSetting* class as shown in 9.7 and identify the *ElementName* field containing *BootSeq* and corresponding *InstanceID* (IPL or UEFI)

B) ENUMERATE the *DCIM_BootSourceSetting* class as shown in 9.9 and identify the boot source *InstanceID*. The *CurrentAssignedSequence* attribute of each instance defines the instance's place in the zero based indexed boot sequence

C) To invoke the ChangeBootOrderByInstanceID() method, extract the instance information from A) and B) and construct the input parameters per Table 32**Error! Reference source not found.**

D) INVOKE ChangeBootOrderByInstanceID() method

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_BootConfigSetting?InstanceID=IPL

E) Examine output parameters per Table 31

F) Apply the pending values (Section 9.5)

G) Allow several minutes for the UEFI to execute the SSIB task, which will be followed by a reboot of the system

H) Repeat B) to confirm successful execution of the method

9.11 Enable or disable boot source

A) ENUMERATE the *DCIM_BootConfigSetting* class as shown in 9.7 and identify the *ElementName* field containing *BootSeq* and corresponding *InstanceID* .

B) ENUMERATE the *DCIM_BootSourceSetting* class as shown in 9.9 and identify the boot source *InstanceID*. The *CurrentEnabledStatus* attribute of each instance identifies whether it is enable or disabled

C) To invoke the ChangeBootSourceState() method, extract the instance information from A) and B) and construct the input parameters per Table 30

D) INVOKE ChangeBootSourceState() method

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_BootConfigSetting?InstanceID=IPL

E) Examine output parameters per Table 29

F) Apply the pending values (Section 9.5)

G) Allow several minutes for the UEFI to execute the SSIB task, which will be followed by a reboot of the system

H) Repeat B) to confirm successful execution of the method

9.12 One time boot

A) ENUMERATE the *DCIM_BootSourceSetting* class as shown in 9.9 and identify the desired one time boot source: vFlash, IPL, or UEFI

B) To invoke the ChangeBootOrderByInstanceID() method, extract the instance information from A) and construct the input parameters per Table 32; providing an xml with one and only one DCIM_BootSourceSetting InstanceID**Error! Reference source not found.**

C) `INVOKE` ChangeBootOrderByInstanceID() method

Class URI:

http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_BootConfigSetting?InstanceID=OneTime

D) Examine output parameters per Table 31

E) `ENUMERATE` the *DCIM_BootSourceSetting* class as shown in 9.9 and verify that an additional entry for that DCIM_BootSourceSetting appears prefixed with “OneTime:”

F) `ENUMERATE` the *DCIM_BootConfigSetting* class as shown in 9.7, the *OneTime* entry should have an *IsNext* value of 3, which means “Is Next for Single Use”

ANNEX A
(informative)

Related MOF Files

Dell Tech Center MOF Library:

http://www.delltechcenter.com/page/DCIM.Library.MOF

Related Managed Object Format (MOF) files:

DCIM_BootConfigSetting.mof

DCIM_BootSourceSetting.mof

DCIM_BIOSEnumeration.mof

DCIM_BIOSInteger.mof

DCIM_BIOSService.mof

DCIM_BIOSString.mof

DCIM_LCElementConformsToProfile

DCIM_LCRegisteredProfile