Dell Command | Monitor Version 9.0
User's Guide

Notes, Cautions, and Warnings

NOTE: A NOTE indicates important information that helps you make better use of your computer.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

2014 - 09

Rev. A00

Contents

1

Introduction

Client Instrumentation refers to software applications that enable remote management of a client system. The Dell Command | Monitor (Command | Monitor) software application enables remote management using application programs to access the enterprise client system information, monitor the status, or change the state of the system such as remotely shutting down the system. Command | Monitor uses key system parameters through standard interfaces allowing administrators to manage inventory, monitor system health, and gather information of deployed enterprise client systems. This document provides an overview of Dell Command | Monitor and its features.

NOTE: Dell Command | Monitor was formerly Dell OpenManage Client Instrumentation (OMCI). After the OMCI version 8.2.1, OMCI is rebranded as Dell Command | Monitor.

Dell Command | Monitor overview

Command | Monitor manages client systems using the Common Information Model (CIM) standard and Simple Network Management Protocol (SNMP), which are management protocols. This reduces the total cost of ownership, improves security, and provides a holistic approach to manage all the devices including clients, servers, storage, network, and software devices.

Using CIM you can access Command | Monitor through Web Services for Management Standards (WSMAN).

Command | Monitor contains the underlying driver set that collects client system information from different sources including the BIOS, CMOS, System Management BIOS (SMBIOS), System Management Interface (SMI), operating system, Application programming interface (APIs), Dynamic-link library (DLLs), and registry settings. Command | Monitor fetches this information through the CIM Object Manager (CIMOM) interface, Windows Management Instrumentation (WMI) stack or SNMP agent.

Command | Monitor enables IT administrators to remotely collect asset information, modify CMOS settings, receive proactive notifications about potential fault conditions, and get alerts for potential security breaches. These alerts are available as events in event log, CIM Indication or received as SNMP traps after importing the Management Information Base (MIB) file and monitoring it.

Command | Monitor is used to gather asset inventory from the system including BIOS settings, through CIM implementation or SNMP agent. It can be integrated into a console such as Microsoft System Center Configuration Manager by directly accessing the CIM information, or through other console vendors who have implemented the Command | Monitor integration. Additionally, you can create custom scripts to target key areas of interest. You can use these scripts to monitor inventory, BIOS settings, and system health.

NOTE: Default installation does not enable SNMP support. For more information on enabling SNMP support, see *Dell Command | Monitor Installation Guide* at **dell.com/clientsystemsmanagement**.

Command | Monitor architecture

Command | Monitor data provider collects the system information data and stores the information in the proprietary Extensible Markup Language (XML) format. The data manager is a service that loads these providers based on request. Command | Monitor CIM provider layer abstracts the interface to different CIMOM implementations. The input is a combination of XML and Extensible Stylesheet Language (XSL) data in proprietary form, while the output is a CIM object instance based on Management Profiles. The WSMAN that serves as the channel protocol, requests the data from CIMOM and transmits it to the console.

The Command | Monitor architecture has several layers that are integrated with the Microsoft Windows Management Instrumentation (WMI) stack:

- WMI application layer — Consists of management applications, standards-based management tools, and WMI applications such as Microsoft SMS, LANDesk, and WMI Tools. The applications in this layer are consumers of the system's management data supplied by Command | Monitor. These applications request client information and send alerts through WSMAN/CIM Object Manager (CIMOM).
- WMI CIM Provider — Is available under CIMOM and contains two CIM providers, which are registered with the CIMOM:
  - The instance or method provider implements an interface that enables utility operations such as create, delete, modify, and query.
  - The indication provider implements an interface for WMI indications (events).

    When the CIMOM receives a request for information, it routes the request to the appropriate provider. All providers exist in this layer, and they provide information on system devices. The providers send management application requests from the CIMOM to the data router.
- Data Manager — Loads data provider based on request from the upper layer.
- Data Provider — Collects system information like hardware, drivers, and operating system data , and stores them in the proprietary XML format.

The Command | Monitor architecture has several additional layers that are integrated with the SNMP stack:

- SNMP agent — Shows the data received from the data manager as SNMP tables and traps.
- MIB — The MIB files store information about SNMP tables, its attributes and the available traps.

For example, a management console in the WMI application layer requests the available processor information on a client system. The WMI application layer makes the request over the network to the CIMOM on the client system. The CIMOM passes the request to the Command | Monitor CIM provider and data manager. The data manager loads the corresponding data provider, which receives the information and stores it in a proprietary format. The information is then returned (through the same path in reverse) to the management console.

What's new in this release

- Dell OpenManage Client Instrumentation (OMCI) is rebranded to Dell Command | Monitor
- Support for monitoring and alerting for LSI Redundant Array of Independent Disks (RAID) controller
- Support for monitoring and alerting for all sensor probes
- Legacy namespaces **(root/dellomci)** are not supported

- Support for new SNMP 10909 MIB
- Support for the following new tokens:
  - GPS Radio
  - Keyboard Backlight with AC
  - Back Camera
  - <Fn> Lock
  - <Fn> Lock Mode
  - Onboard Unmanaged Network Interface Card (NIC)
  - Unmanaged NIC
  - Rear USB Ports
  - Side USB Ports
  - Trusted Execution
- Support for additional values, **medium_high** and **medium_low** for **fanspeed** token.

NOTE: Legacy namespaces and SNMP 10892 MIB have been removed.

For more information on tokens, see the *Dell Command | Monitor Reference Guide* at **dell.com/clientsystemsmanagement**.

Features

The key features of Command | Monitor are:

- CIM 2.17 schema support
- BIOS configuration
- WMI security
- Event reporting
- Remote shutdown
- Access to system information using WMI-CIM protocol, WSMAN, and SNMP
- Compilation of detailed asset information
- Remote wake-up configurability
- Remote modification of system settings
- Monitoring of system health and reports status
- RAID monitoring and alerting for Intel and LSI controllers.
- SNMP monitoring and traps

CIM 2.17 schema support

Command | Monitor conforms to the CIM 2.17 Schema, and includes two WMI providers:

- WMI Indication Provider or Polling Agent
- WMI Instance or Method Provider

BIOS configuration

Command | Monitor provides the ability to configure a system BIOS, including management of its boot order.

WMI security

WMI provides user authentication before granting access to the CIM data and methods. Access privileges are enforced by Distributed Component Object Model (DCOM) security and CIMOM. Complete or limited access, is granted to users on per-namespace basis. There is no class implementation or property-level security. By default, users who are members of the administrators group have complete local and remote access to WMI.

You can configure WMI security using the WMI Control available in the Computer Management console under the Services and Applications section. Right-click **WMI Control**, and then click **Properties**. You can configure namespace-specific security from the **Security** tab. You can also run **WMI Control** from the **Start** menu or from the **CLI**, by running `wmimgmt.msc`.

Event reporting

Command | Monitor detects events on Dell systems and alerts the local user and network administrator about potential failures, configuration changes, and chassis intrusions. These events are displayed by a systems management application such as OpenManage Essentials (OME).

Remote shut down

Command | Monitor supports remote system shut down and reboot.

Information access

Command | Monitor provides access to system information such as BIOS revision and system model through WMI using CIM. The WSMAN protocol can also be used to access this information through WMI.

Detailed asset information

Command | Monitor provides access to detailed inventory information such as processors, PCI devices, and batteries.

Remote wake-up configuration

Command | Monitor supports configuration of remote wake-up settings. Remote wake-up is a function of the client system and Network Interface Card (NIC).

Remote modification of system settings

Command | Monitor allows administrators to retrieve and set business client BIOS settings such as USB port configuration, boot order, and NIC settings.

System health and status

Command | Monitor monitors the system health such as fan status, and reports the status through NT event log entries and CIM events.

RAID monitoring and alerting for Intel and LSI Controllers

Monitoring and alerting for Intel and LSI RAID controllers for its physical and logical drives.

SNMP monitoring and traps

Command | Monitor confirms to SNMP v1 and supports monitoring of system attributes and traps.

2

Standards and protocols

Command | Monitor uses Microsoft Windows Management Instrumentation (WMI) and enables Web Services-Management (WSMAN) protocols. Command | Monitor uses Simple Network Management Protocol (SNMP) to describe several variables of the system.

CIM, SNMP, WMI, and WSMAN technology overview

The Desktop Management Task Force (DMTF) is the industry-recognized standards body that leads the development, adoption, and unification of management standards (including CIM and ASF), and initiatives for desktop, enterprise, and internet environments.

CIM

The CIM, created by the DMTF as part of the Web-based Enterprise Management (WBEM) initiative, provides a unified view of physical and logical objects in the managed environment.

The following are important CIM details:

- CIM is an object-oriented data model for describing management information. CIM describes the way the data is organized, not necessarily the transport model used to transport the data. The most prevalent transport method is WMI.
- CIM-capable management applications gather information from a variety of CIM objects and devices, including client and server systems, network infrastructure devices, and applications.
- The CIM specification details mapping techniques for improved compatibility with other management protocols.
- The CIM data model abstracts and describes all elements in a network environment. The CIM schema provides the actual data model descriptions and arranges the network into a series of managed objects, all interrelated and broadly classified.
- The CIM schema is defined by the Managed Object Format (MOF) file, which provides a standardized model for describing management information between clients in a management system. The MOF file is not bound to a particular implementation, and it allows the interchange of management information among different management systems and clients.

SNMP

SNMP is a widely accepted solution to manage devices on Internet Protocol (IP) networks. SNMP is developed and maintained by Internet Engineering Task Force (IETF). Command | Monitor accesses information and monitor client systems using SNMP. Devices that typically support SNMP include routers, switches, servers, workstations, most of the hardware components, and so on. It consists of a set of standards for network management, including an application layer protocol, a database schema, and a set of data objects. SNMP exposes management data in the form of variables on the managed systems, which describe the system configuration. These variables can then be queried by managing applications.

SNMP does not define what information (which variables) a managed system should offer. Rather, SNMP uses an extensible design, where the list of available information is defined by management information

bases (MIBs). MIBs describe the structure of the management data of a device and its subsystems. MIBs use a hierarchical namespace containing object identifiers (OID). Each OID identifies a variable that can be read through SNMP.

WMI

WMI is the Web-based Enterprise Management (WBEM) effort implementation by Microsoft. It is implemented on the Microsoft Windows platforms. WMI supports CIM and Microsoft-specific CIM extensions.

WMI includes:

- A powerful set of native services such as query-based information retrieval and event notification.
- Extensive scripting capabilities through the Windows Scripting Host (WSH).
- The CIMOM, which is the interface and manipulation point for CIM objects and information.
- The repository, where CIMOM stores management data.

In the Command | Monitor architecture, CIMOM and the repository are represented by the Microsoft WMI Object Manager. The CIMOM is the interface and manipulation point for CIM objects and information. CIMOM acts as a facilitator in gathering information and manipulating object properties. Microsoft has implemented this component as the Windows management (winmgmt) service. The CIMOM is a software middle layer that mediates interactions between high-level management applications and the low-level instrumentation, such as Command | Monitor, and other providers. The CIMOM ensures that data supplied by providers is presented to management applications in a uniform and provider-independent way. The CIMOM does this by using the Component Object Model (COM) Application Programming Interface (API).

The repository is a binary file where the CIMOM stores management data. The data includes information from the compiled Managed Object Format (MOF) file(s), including the CIM class definitions, properties, qualifiers, and hierarchical relationships. Instance data, as it becomes available, is also stored in the repository.

WMI provides a scripting interface. Using VBScript or JScript, you can write scripts, connect to WMI services locally or remotely, retrieve information, or run methods. You can script most of the Command | Monitor tasks as Command | Monitor is implemented through WMI.

For more information on VBScript and sample scripts, see the *Dell Command | Monitor Reference Guide* available at **dell.com/clientsystemsmanagement**.

For more information on WMI, see **technet.microsoft.com**.

NOTE: To connect remotely to WMI services, you must have Administrator privileges on both the local and remote systems.

WSMAN

The WSMAN protocol is a DMTF open standard defining a Simple Object Access Protocol (SOAP)-based protocol for managing servers, devices, applications, and web services. It uses data from CIMOM to facilitate the management.

WSMAN is a protocol that provides an abstraction layer to access the CIM information. The reason is that the console can use WSMAN to communicate with in-band or out-of-band systems to gather asset

inventory and to set information or run methods. In in-band systems, the WSMAN layer also abstracts the operating system present underneath. However, Command | Monitor does not require WSMAN and it does not directly enable WSMAN as it is only a protocol.

For more information on managing WSMAN from DMTF, see **dmtf.org/standards/wsman**.

For more information on enabling WSMAN based management of WMI on a system running the Windows operating system, see **msdn.microsoft.com/en-us/library/aa384426%28v=VS.85%29.aspx**.

For more information on the DMTF profiles used in Command | Monitor, see *Dell Command | Monitor Reference Guide* at **dell.com/clientsystemsmanagement**.

PowerShell

Windows PowerShell is a task automation and configuration management framework from Microsoft. PowerShell consists of a command-line shell and associated scripting language built on the .NET Framework. PowerShell provides full access to COM and WMI, enabling administrators to perform administrative tasks such as configuration and monitoring on both local and remote systems running the Windows operating system using the services of Command | Monitor.

Administrators can write custom PowerShell scripts (files suffixed by **.ps1**) that connect to the DCIM namespace and allows monitoring custom actions on the system.

3

System requirements

This chapter provides information about the hardware and software requirements of Command | Monitor.

Hardware requirement

Requirement	Details
System	Enterprise client system with SMBIOS 2.3 or later.

Software requirements

Requirement	Details
Supported operating system	• Microsoft Windows 8.1 • Microsoft Windows 8 • Microsoft Windows 7 • Microsoft Windows Vista
Supported framework	• Microsoft .NET 4.0

4

User scenarios

This chapter describes the various user scenarios of Command | Monitor.

You can use Command | Monitor for:

- Asset management
- Configuration management
- Health monitoring
- Profiles

Scenario 1: Asset management

A company that uses many Dell systems was not able to maintain accurate inventory information because of changes in the business and IT staff. The Chief Information Officer (CIO), requests a plan for identifying the systems that can be upgraded to Microsoft Windows latest versions. This requires an assessment of the deployed systems to determine the size, scope, and financial impact of such a project. The information collection involves a significant effort. Deploying IT staff to each client system is expensive, in terms of man-hours and end-user interruptions.

Using Command | Monitor on each Dell system, the IT manager can quickly collect information remotely. Using tools such as Microsoft System Center Configuration Manager (SCCM), the IT manager queries each client system over the network and collects information such as CPU type and speed, memory size, hard-drive capacity, BIOS version, and current operating system version. Once the information is collected, it can be analyzed to determine the systems that can be upgraded to Windows latest versions.

You can also get asset inventory through script or any Windows Management Instrumentation (WMI) command line.

SCCM integration

You can integrate SCCM with Command | Monitor by:

- Using the MOF file within Command | Monitor install package, which contains all the Command | Monitor classes and importing to ConfigMgr

  The MOF is located at:

  C:\Program Files\Dell\Command_Monitor\ssa\omacim\OMCI_SMS_DEF.mof
- Extending asset reporting capabilities using collections

Scenario 2: Configuration management

A company plans to standardize the client platform and manage each system through its lifecycle. As part of this effort, the company acquires a suite of tools and plans to automate the deployment of a new client operating system using the Preboot Execution Environment (PXE).

The challenge is to modify the boot order setting in the BIOS of each client computer without manually visiting the desktop. With Command | Monitor installed on each client system, the IT department of the company has several options for remotely modifying the boot order. The OpenManage Essentials (OME) is a management console that can be used to remotely monitor BIOS settings on all Enterprise Client Systems. Another option is to write a script (VB/PowerShell/WMIC) that changes the boot order setting. The script can be remotely delivered over the network and run on each client system.

For more information on Command | Monitor, see *Dell Command | Monitor Reference Guide* at **dell.com/clientsystemsmanagement**.

Standardized configurations can provide significant cost savings for companies of all sizes. Many organizations deploy standardized client systems, but few manage the system configuration throughout the life of the computer. With Command | Monitor installed on each client system, the IT department can lock down Legacy ports to prevent the use of unauthorized peripherals, or enable Wake On LAN (WOL) to revive the system from a sleep state during non-peak hours to perform systems management tasks.

Scenario 3: Health monitoring

A user receives read error messages while trying to access certain files on the client-system hard drive. The user reboots the system and the files now appear to be accessible. The user disregards the initial problem because it appears to have resolved itself. Meanwhile, Command | Monitor queries the hard drive with the problem for a predicted failure and passes a Self-Monitoring, Analysis and Reporting Technology (SMART) alert to the management console. It also displays the SMART error to the local user. The alert indicated that several read/write errors are occurring in the hard drive. The IT department of the company recommended that the user must take a backup of critical data files immediately. A service technician is dispatched with a replacement drive.

The hard drive is replaced before it failed, preventing user downtime, a help desk call, and a technician trip to the desktop to diagnose the problem.

Monitoring system events through Windows Event Viewer or CIM indication

Command | Monitor supports monitoring events through the following procedures:

- Pulling the log through WMI class **DCIM_LogEntry**.
- Monitoring CIM indication through **DCIM_AlertIndication** class.
- Monitoring events through Simple Network Management Protocol (SNMP).

For more information on Command | Monitor, see *Dell Command | Monitor Reference Guide* at **dell.com/clientsystemsmanagement**.

Scenario 4: Profiles

IT administrators are required to manage client system in multi-vendor and distributed enterprise environments. They face challenges as they must master diverse set of tools and applications while managing several desktop and mobile client systems in various networks. To reduce the cost of these requirements and represent the provided management data, the industry-standard Distributed Management Task Force (DMTF) and Data Center Infrastructure Management (DCIM-OEM) profiles are implemented in Command | Monitor. Some of the DMTF profiles are explained in this guide.

For more information on Command | Monitor, see *Dell Command | Monitor Reference Guide* at **dell.com/clientsystemsmanagement**.

Battery profile

- Determine the status of the battery by enumerating or getting the instance of the class **DCIM_Battery**.
- Determine the estimate run time and see the estimated remaining charge.
- Check if the health information of the battery can be determined using the properties *Operational Status* and *HealthState* of the class **DCIM_Battery**.
- Get additional information about the health of a battery using **DCIM_Sensor.CurrentState** property or the **CIM_NumericSensor.CurrentState** property.

BIOS management profile

- Determine the BIOS version by enumerating the instance of the class **DCIM_BIOSElement.**.
- Check whether BIOS attribute value can be modified or not. Get the instance of the class, **DCIM_BIOSEnumeration**. The attribute can be modified if the property **IsReadOnly** is set to FALSE.
- Set the system password (SystemPwd). Run the **DCIM_BIOSService.SetBIOSAttributes()** method and set the SystemPwd to AttributeName and password value to AttributeValue parameters.
- Set the BIOS or Admin password (AdminPwd). Run the **DCIM_BIOSService.SetBIOSAttributes()** method and set the AdminPwd to AttributeName and password value to AttributeValue parameters.
- Run the **DCIM_BIOSService.SetBIOSAttributes()** method and specify the AttributeName and AttributeValue parameters.
- To modify a BIOS Attribute when BIOS or Admin password is set, run the **DCIM_BIOSService.SetBIOSAttributes()** method and specify the AttributeName, AttributeValue, and current BIOS password as the AuthorizationToken input parameter.

Boot control

- Change the sequence of the boot items in the Legacy and UEFI boot list.
- Enable or disable the boot items in the Legacy and UEFI boot list.
- Find the current boot configuration by enumerating the instances of the class **DCIM_ElementSettingData** whose **IsCurrent** property is set to **1**. The **DCIM_BootConfigSetting** represents the current boot configuration.

Base desktop mobile

- Determine the system model, service tag, and serial number by enumerating the instance of the class, **DCIM_ComputerSystem**.
- Run the **DCIM_ComputerSystem.RequestStateChange()** method and set the RequestedState parameter value to **3**. Turn off the system.

- Reboot the system. Run the **DCIM_ComputerSystem.RequestStateChange()** method and set the **RequestedState** parameter value to **11**.
- Determine the power state of the system.
- Determine the number of processors in the system by querying **DCIM_Processor**, instances which are associated with the Central Instance through the **DCIM_SystemDevice** association.
- Get the system time. Run the **DCIM_TimeService.ManageTime()** method and set the **GetRequest** parameter to **True**.
- Check the health status of the managed element.

Log record

- Identify the log name by selecting the **DCIM_RecordLog** instance in which the **ElementName** property corresponds to the log name.
- Find the individual log entries. Get all the instances of **DCIM_LogEntry** that are associated with the given instance of **DCIM_RecordLog** through the **DCIM_LogManagesRecord** association. Sort the instances based on the **RecordID**.
- Check whether record logs are enabled or not by enumerating the instance of the class **DCIM_RecordLog** whose property **Enabledstate** is set to **2** (represents enabled) and **EnabledState** is set to **3** (represents disabled).
- Sort the log records based on the time stamp of the log entry. Get all the instances of **DCIM_LogEntry** that are associated with the given instance of **DCIM_RecordLog** through the **DCIM_LogManagesRecord** association. Sort the instances of **DCIM_LogEntry** based on the **CreationTimeStamp** property value in Last In First Out (LIFO) order.
- Clear logs by running the **ClearLog()** method for the given instance of the **DCIM_RecordLog**.

Physical asset

- Obtain the physical inventory for all the devices in a system.
- Obtain the physical inventory for a system chassis.
- Determine the part number of a failing component.
- Determine whether the slot is empty or not.

System memory profile

- Obtain the memory information of the system.
- Obtain the physical memory information of the system.
- Check the system memory size.
- Check the available system memory size.
- Check the physical system memory size.
- Check the health status of system memory.

5

Using Dell Command | Monitor

You can view the information provided by Command | Monitor by accessing:

- root\dcim\sysman (standard)

Command | Monitor provides the information through classes in these namespaces.

For more information on the classes, see *Dell Command | Monitor Reference Guide* at **dell.com/clientsystemsmanagement**.

Polling interval setting

You can change the polling interval of fan probe, temperature probe, voltage probe, current probe, disk capacity increase/decrease, memory size increase/decrease and number of processors increase/decrease, using the files **dcsbdy32.ini** or **dcsbdy64.ini**. The **dcsbdy32/64.ini** file is present at the following location:

<Command | Monitor installed location>\omsa\ini

NOTE: The numbers in the INI file are multiples of **23**. The default polling interval for disk capacity and Self-Monitoring, Analysis and Reporting Technology (SMART) alert is **626** seconds (the real time = 626 X 23 seconds which is approximately 3 hours).

RAID status reporting

Command | Monitor enables the RAID configuration information and monitors the RAID functionality for client systems with hardware and driver support. You can use RAID classes to receive the details about RAID levels, driver information, controller configuration, and controller status. After the RAID configuration is enabled, you can receive alerts for degradation or failure of drives and controllers.

NOTE: RAID status reporting is supported only for the RAID controllers which work on Common Storage Management Interface (CSMI) version 0.81 compliant drivers. OMCI 8.1 and later versions support monitoring only on the Intel on-chip RAID controller; and from OMCI 8.2 and later versions support Alerting for Intel on-chip RAID controller. Command | Monitor 9.0 and later versions also support LSI controller for monitoring and alerting feature.

Monitoring the client systems

Command | Monitor supports Simple Network Management Protocol (SNMP) for monitoring and managing client systems such as notebooks, desktops, and workstations. The Management Information Base (MIB) file is shared between Command | Monitor and Server Administrator.

Command | Monitor from version 9.0 has been modified to use an OID that is specific to client OID (10909) for consoles to identify client systems.

For more information on SNMP, see *Dell Command | Monitor SNMP Reference Guide* at **dell.com/clientsystemsmanagement**.

Detecting advance format drives

Client systems are transitioning to Advanced Format (AF) drives for larger storage capacity and to address the limitations of 512-byte sector hard drives (HDDs). The hard drives transitioning to 4KB sectors maintain backward compatibility, while the current AF hard drive, known as 512e hard drive, match 512-byte SATA and operate at 4KB. During the transition, you may encounter performance issues such as, partition mis-aligned drives in the client systems resulting in failure of sector-based encryption software packages that handle 512e drives. Command | Monitor allows you to identify if the hard drive on a system is 4KB AF drive, which helps to prevent these issues.

Boot configurations

A client system can have two boot configurations:

- Legacy (BIOS)
- UEFI

In Dell Command | Monitor, the boot configuration (Legacy or UEFI) is modeled using the following classes:

- **DCIM_ElementSettingData**
- **DCIM_BootConfigSetting**
- **DCIM_OrderedComponent**
- **DCIM_BootSourceSetting**

NOTE: The terms "Boot Configuration" and "Boot List Type" are used interchangeably and convey the same meaning representing Legacy or UEFI.

DCIM_BootConfigSetting

An instance of **DCIM_BootConfigSetting** represents a boot configuration that is used during the boot process. For example, on client systems, there are two types of boot configurations — Legacy and UEFI. So, **DCIM_BootConfigSetting** has a maximum of two instances to represent, one each for Legacy and UEFI.

You can determine if the **DCIM_BootConfigSetting** represents Legacy, using the following properties:

- InstanceID = "DCIM:BootConfigSetting:Next:1"
- ElementName = "Next Boot Configuration Setting : Boot List Type 1"

You can determine if the **DCIM_BootConfigSetting** represents UEFI, using the following properties:

- InstanceID = "DCIM:BootConfigSetting:Next:2"
- ElementName = "Next Boot Configuration Setting : Boot List Type 2"

DCIM_BootSourceSetting

This class represents the boot devices or sources. The **ElementName**, **BIOSBootString**, and **StructuredBootString** properties contain a string that identifies the boot devices. For example, floppy,

hard disk, CD/DVD, network, Personal Computer Memory Card International Association (PCMCIA), Battery Electric Vehicle (BEV), or USB. Based on the boot list type of the device, an instance of **DCIM_BootSourceSetting** is associated with one of the instances of **DCIM_BootConfigSetting**.

DCIM_OrderedComponent

The **DCIM_OrderedComponent** association class is used to associate instances of **DCIM_BootConfigSetting** with instances of **DCIM_BootSourceSetting** representing one of the boot list type (Legacy or UEFI) , which the boot devices belongs to. The **GroupComponent** property of **DCIM_OrderedComponent** refers to the **DCIM_BootConfigSetting** instance and the **PartComponent** property refers to the **DCIM_BootSourceSetting** instance.

Changing boot sequence using the ChangeBootOrder method

To change the boot sequence follow the steps:

1. Check for the boot list type using:
   - WMIC Command:`wmic /namespace:\\root\dcim\sysman path dcim_BootConfigSetting get ElementName/format:list`
   - PowerShell Command: `gwmi -namespace root\dcim\sysman -class dcim_BootConfigSetting -Property ElementName`
2. Check for boot order type (Legacy or UEFI) using:
   - WMIC Command: `wmic /namespace:\\root\dcim\sysman path dcim_ElementSettingData.IsCurrent=1 get SettingData /format:list`
   - PowerShell Command: `gwmi -namespace root\dcim\sysman -class dcim_elementSettingData -Filter "IsCurrent=1" -Property SettingData`
3. Change the boot order using:
   - WMIC Command: `wmic /namespace:\\root\dcim\sysman path dcim_bootconfigsetting call ChangeBootOrder /?:full`
   - PowerShell Command: `(gwmi -namespace root\dcim\sysman -class dcim_bootconfigsetting).getmethodparameters("ChangeBootOrder")`

   The arguments required for ChangeBootOrder method are:
   - Authorization Token — This is the Administrator or boot password.
   - Source — This is the boot order list taken from DCIM_OrderedComponent.PartComponent property. The new boot order is determined by the order of boot devices in the **source** array.

Setting BIOS attributes

In Dell Command | Monitor, the following methods are added for changing the system settings and state of the local or remote systems:

- **SetBIOSAttributes** — For changing the BIOS setting
- **ChangeBootOrder** — For changing the boot configuration
- **RequestStateChange** — For shutting down and restarting the system
- **ManageTime** — Returns system time

You can run these methods using winrm, VB script, PowerShell commands, wmic, wbemtest.exe, and WMI wbemtest.

You can set BIOS attributes using the SetBIOSAttributes method. The procedure is explained below by enabling the Trusted Platform Module (TPM) as an example.

NOTE: Make sure the TPM option is cleared in the BIOS before following the procedure to enable the TPM.

To enable TPM:

1. Set the BIOS password on the system if not set already using the following PowerShell command:

   ```
   (gwmi -Class DCIM_BIOSService -Namespace root\dcim
   \sysman).SetBIOSAttributes($null,$null,"AdminPwd","enter a new password")
   ```

2. To enable TPM security use the following command and restart the system after it:

   ```
   (gwmi -Class DCIM_BIOSService -Namespace root\dcim
   \sysman).SetBIOSAttributes($null,$null,"Trusted Platform
   Module","1","provide the password")
   ```

3. To activate the TPM use the following command and again restart the system:

   ```
   (gwmi -Class DCIM_BIOSService -Namespace root\dcim
   \sysman).SetBIOSAttributes($null,$null,"Trusted Platform Module
   Activation","2","provide the password")
   ```

4. Restart the system.

   NOTE: Use PowerShell with Administrator privileges.

6

Frequently asked questions

How to find the boot order (sequence) of the boot configuration using DCIM_OrderedComponent.AssignedSequence property?

When a **DCIM_BootConfigSetting** instance (Legacy or UEFI) has multiple **DCIM_BootSourceSetting** instances (boot devices) associated with it through instances of the **DCIM_OrderedComponent** association, the value of the **DCIM_OrderedComponent.AssignedSequence** property is used to determine the sequence in which the associated **DCIM_BootSourceSetting** instances (boot devices) are used during the boot process. A **DCIM_BootSourceSetting**, whose associated **CIM_OrderedComponent.AssignedSequence** property is equal to **0** is ignored and not considered part of the boot order.

How to change the boot order?

The boot order can be changed using the **DCIM_BootConfigSetting.ChangeBootOrder()** method. The **ChangeBootOrder()** method sets the order in which the instances of **DCIM_BootSourceSetting** are associated with a **DCIM_BootConfigSetting** instance. The method has one input parameter; **Source**. The **Source** parameter, is an ordered array of **PartComponent** property from **DCIM_OrderedComponent** class that represents the association between **DCIM_BootSourceSetting** instances (boot devices) and **DCIM_BootConfigSetting** instance (boot list type-Legacy or UEFI).

How to disable boot devices?

On changing the boot order, the value of the **AssignedSequence** property on each instance of **DCIM_OrderedComponent**, that associates the target **DCIM_BootConfigSetting** instance with a **DCIM_BootSourceSetting** instance that is not present in the input array of **Source** parameter, is set to **0**, which indicates that the device is disabled.

Fail login message appear when connect to namespace with wbemtest. How can I overcome that?

Launch **wbemtest** with Administrator privilege level to overcome any login message. Go to the Internet Explorer from the **All Programs** list, right-click and **Run as administrator** to start the **wbemtest** and avoid any namespace oriented error.

How do I run TechCenter scripts without any issues?

The following are the prerequisites while executing the VBS scripts provided in Command | Monitor Techcenter link:

1. Please configure **winrm** on the system using the command `winrm quickconfig`.
2. Check if the token support exists on the system by referring to:
   - The **F2 Screen** in BIOS Setup.
   - Using tool like **wbemtest** to check the key value define in the script to be existing on the system.

   NOTE: Dell recommends using the latest BIOS available at **dell.com/support**.

How to set the BIOS attributes?

BIOS Attributes can be changed using the **DCIM_BootService.SetBIOSAttributes()** method. The **SetBIOSAttributes()** method sets the value of the instance defined in the **DCIM_BIOSEnumeration** class. The method has seven input parameters. The first two parameters can be empty or null. The third parameter **AttributeName** needs to take the input mapping to the value of attribute name instance of **DCIM_BIOSEnumeration** class. The fourth parameter or **AttributeValue** can be any of the possible values of the Attribute Name as defined in the**DCIM_BIOSEnumeration** class. If the BIOS Password is set on the system, then you have to provide the same in the fifth argument. The sixth and seventh argument can again be empty or null.

7

Troubleshooting

Unable to remotely connect to Windows Management Instrumentation

If Common Information Model (CIM) information for a remote client computer system is not available to the management application or if a remote BIOS update that uses Distributed Component Object Model (DCOM) fails, the following error messages are displayed:

- **Access Denied**
- **Win32:RPC server is unavailable**

1. Verify that the client system is connected to the network. Type the following in the command prompt of the server:

   `ping <Host Name or IP Address>` and press `<Enter>`.
2. Perform the following step if both the server and the client system are in the same domain:
   - Verify that the domain administrator account has Administrator privileges for both systems.

   Perform the following step if both the server and the client system are in a workgroup (not in the same domain):
   - Make sure that the server is running on the latest Windows Server.

   NOTE: Back up your system data files before changing the registry. Editing the registry incorrectly may render your operating system unusable.
3. Edit the registry change on the client system. Click **Start** → **Run**, then type **regedit**, and then click **OK**. In the **Registry Editor** window, browse to **My Computer\HKEY_LOCAL_MACHINE\SYSTEM \CurrentControlSet\Control\Lsa**.
4. Set the **forceguest** value to **0** (default value is **1**). Unless you modify this value, the user remotely connecting to the system will have guest privileges, even if the supplied credentials provide Administrator privileges.
   a. Create an account on the client system with the same user name and password, as an administrator account on the system running the WMI management application.
   b. If you are using IT Assistant, run the IT Assistant ConfigServices utility (**configservices.exe** in the**/bin** directory under the IT Assistant installation directory). Configure IT Assistant to run under a local administrator account, which is also now an administrator on the remote client. Also, verify that DCOM and CIM are enabled.
   c. If you are using IT Assistant, use the administrator account to configure subnet discovery for the client system. Enter the user name as *<client machine name>\<account name>*. If the system has already been discovered, remove the system from the list of discovered systems, configure subnet discovery for it, and then rediscover it.

   NOTE: Dell recommends using Dell OpenManage Essentials as replacement for IT Assistant. For more information on Dell OpenManage Essentials see, **dell.com/clientsystemsmanagement**.
5. Perform the following steps to modify user privilege levels for connecting remotely to a system's WMI:

a. Click **Start** → **Run**, type `compmgmt.msc`, and then click **OK**.
b. Browse to **WMI Control** under **Services and Applications**.
c. Right-click **WMI Control**, and then click **Properties**.
d. Click the **Security** tab and select **DCIM/SYSMAN** under the **Root** tree.
e. Click **Security**.
f. Select the specific group or user that you want to control access and use the **Allow** or **Deny** check box to configure the permissions.

6. Perform the following steps to connect to a WMI (**root\DCIM/SYSMAN**) on a system from a remote system using WMI CIM Studio:
   a. Install **WMI tools** along with **wbemtest** on the local system, and then install Dell Command | Monitor on the remote system.
   b. Configure the firewall on the system for WMI remote connectivity. For example, open the TCP ports 135 and 445 in Windows firewall.
   c. Set the **Local Security** setting to **Classic - local users authenticate as themselves for Network access: Sharing and security model for local accounts** in the **Local Security Policy**.
   d. Connect to the WMI (**root\DCIM/SYSMAN**) on the local system from a remote system using WMI wbemtest. For example, \\[Target remote system IP Address]\root\DCIM/SYSMAN
   e. Enter the Administrator credentials of the target remote system if prompted.

   For more information on WMI, see the applicable Microsoft documentation at **msdn.microsoft.com**.

Installation failure

If you are unable to complete Dell Command | Monitor installation, ensure that:

- You have Administrator privilages on the target system.
- The target system is a Dell manufactured system with SMBIOS version 2.3 or later.

NOTE: To check the SMBIOS version on the system, go to **Start** → **Run** and run the **msinfo32.exe** file and check for the SMBIOS version in System Summary page.

NOTE: The system must be running supported Microsoft Windows operating system.

NOTE: The sytsem has to be upgraded to .NET 4.0 or later versions.

8

Contacting Dell

NOTE: If you do not have an active Internet connection, you can find contact information on your purchase invoice, packing slip, bill, or Dell product catalog.

Dell provides several online and telephone-based support and service options. Availability varies by country and product, and some services may not be available in your area. To contact Dell for sales, technical support, or customer service issues:

1. Go to **dell.com/support**.
2. Select your support category.
3. Verify your country or region in the **Choose a Country/Region** drop-down list at the bottom of the page.
4. Select the appropriate service or support link based on your need.

Other documents you may need

In addition to this User's Guide, you can access the following documents at **dell.com/clientsystemsmanagement**. Click Command Monitor (formerly OpenManage Client Instrumentation) and then click the appropriate product version link in **General support** section.

- The *Dell Command | Monitor Reference Guide* provides detailed information on all Client Instrumentation classes, properties, and descriptions.
- The *Dell Command | Monitor Installation Guide* provides information on installing Client Instrumentation.
- The *Dell Command | Monitor SNMP Reference Guide* provides Simple Network Management Protocol (SNMP) Management Information Base (MIB) applicable to Dell Command | Monitor.

Accessing documents from Dell support site

You can access the required documents in one of the following ways:

- Using the following links:
  - For all Enterprise Systems Management documents — **dell.com/softwaresecuritymanuals**
  - For Enterprise Systems Management documents — **dell.com/openmanagemanuals**
  - For Remote Enterprise Systems Management documents — **dell.com/esmmanuals**
  - For OpenManage Connections Enterprise Systems Management documents — **dell.com/OMConnectionsEnterpriseSystemsManagement**
  - For Serviceability Tools documents — **dell.com/serviceabilitytools**
  - For Client Systems Management documents — **dell.com/clientsystemsmanagement**
  - For OpenManage Connections Client Systems Management documents — **dell.com/connectionsclientsystemsmanagement**
- From the Dell Support site:

a. Go to **dell.com/support/home**.

b. Under **General support** section, click **Software & Security**.

c. In the **Software & Security** group box, click the required link from the following:

   - **Enterprise Systems Management**
   - **Remote Enterprise Systems Management**
   - **Serviceability Tools**
   - **Client Systems Management**
   - **Connections Client Systems Management**

d. To view a document, click the required product version.

- Using search engines:

  - Type the name and version of the document in the search box.