

Dell™, Inc.

# **DELL OPENMANAGE CLIENT INSTRUMENTATION**

*Integration for IT Professionals*

Nathan Martell  
Software Engineer, Product Group

## Table of Contents

Executive Summary.....	4
CIM and WMI Technology Overview .....	4
CIM.....	4
WMI .....	5
Client Instrumentation Overview <sup>i</sup> .....	6
OMCI Architecture .....	6
OMCI Alerts.....	7
Sample Scenarios <sup>i</sup> .....	7
Asset Management .....	7
Configuration Management.....	8
Health Monitoring.....	8
Management Application Support <sup>i</sup> .....	8
Enterprise Consoles .....	8
Installing OMCI.....	9
Conclusion <sup>i</sup> .....	9
Using OMCI .....	9
Namespace Information .....	9
Classes.....	10
General System .....	10
Boot devices.....	10
BIOS Config.....	10
HDD .....	10
Power .....	10
BIOS Flash.....	10
Software info.....	11
Hardware .....	11
Sensors.....	12
Other.....	12

Indications.....	13
Appendices.....	14
Appendix A – Sample Scripts.....	14
Clear Chassis Intrusion Status .....	14
Disable Desktop Messages for Fan Alerts.....	15
Disable Desktop Messages for All Alerts.....	17
Enable Trusted Platform Module.....	18
Get/Set Boot Order .....	20
Change/Set the BIOS Admin Password.....	24
Change/Set the BIOS Boot Password.....	25

## Executive Summary

Dell OpenManage Client Instrumentation (OMCI) is software that allows remote management application programs to access information about the client computer, monitor the status of the client computer, or change the state of the computer, such as shutting it down remotely. OMCI uses the Common Information Model (CIM), which is a systems management protocol defined by industry standards. Microsoft® Windows® Management Instrumentation (WMI) is Microsoft's implementation of CIM instrumentation. OMCI provides data to WMI, which is the common interface to WMI management applications.

Although OMCI has numerous features, its primary purpose is to package and provide access to information requested by the CIM Object Manager (CIMOM), which in turn provides the information to systems management application programs such as Dell OpenManage IT Assistant.

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 between many different management systems and clients.

OMCI uses a variety of data sources resident on the client computer:

- Microsoft Win32 WMI providers
- System management basic input/output system (SMBIOS)
- Device drivers
- Operating system registry
- Operating system Application Programming Interfaces (APIs)
- WMI Repository

## CIM and WMI Technology Overview<sup>i</sup>

The Distributed Management Task Force (DMTF) is the industry-recognized standards body that leads the development, adoption, unification of management standards (including CIM, and ASF), and initiatives for desktop, enterprise, and Internet environments. Working with Dell and other key technology vendors and standards groups, the DMTF enables an integrated, cost-effective approach to management. CIM is a key component of this vision.

### CIM

CIM is an object-oriented data model for describing management information. Designed to be implementation-neutral, CIM-capable management applications can gather information from a variety of CIM objects and devices, including client and server systems, network infrastructure devices, and applications.

CIM also helps to detail 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.

Using this approach, CIM goes beyond representing raw data; instead, it describes the enterprise network and allows a much more accurate representation of the network environment and each managed element. There is wide acceptance of CIM by key vendors who have transitioned their products to CIM.

## WMI

CIM support is included in Windows® XP and Windows Vista® via the WMI infrastructure. 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 via the Windows Scripting Host (WSH).
- The CIM Object Manager (CIMOM), which is the interface and manipulation point for CIM objects and information.

The CIMOM acts as a facilitator in gathering information and manipulating object properties. CIM software components (called “providers”) act as Component Object Manager/Distributed Component Object Manager (COM/DCOM) servers that handle requests from the CIMOM when needed.

The CIMOM stores data in an area called the WMI repository. Management objects stored in the WMI repository can be either physical or logical entities. All CIM objects that the CIMOM has knowledge of reside in the repository. Users may contribute to the repository by compiling files in the Managed Object Format (MOF).

A variety of management applications support CIM and WMI. Most can collect information directly from the CIMOM interface, but there are alternate methods such as Open Database Connectivity (ODBC), Active Directory Services Interface (ADSI), and the WMI scripting application program interface (API).

WMI provides for user authentication before granting access to CIM data. Access privileges are enforced by DCOM security and the CIMOM. Full or limited access can be granted to users on a per-namespace basis. There is no implementation of class- or property-level security. By default, users who are members of the administrators group have full local and remote access to WMI. Note: User Account Control in Windows Vista requires an elevation of privileges for local members of the administrators group to have full access to the DellOMCI namespace. See the OMCI User’s Guide for more information.

The WMI scripting interface can retrieve information, execute CIM actions (or “methods”), and run VBScript or Jscript scripts that connect to WMI services locally or remotely. Because OMCI is implemented through WMI, most OMCI tasks can be scripted.

## Client Instrumentation Overview<sup>i</sup>

Client instrumentation is a general term that typically refers to the software agent or application that enables a client computer to be managed remotely. Instrumentation is usually vendor-specific, but based on industry standards (such as CIM) defined by the DMTF.

An industry-standard client interface for collecting information and alerting helps to ensure interoperability with management applications. Each application is not required to understand multiple underlying methodologies for collecting and manipulating client system information or settings on a specific device.

OMCI is the Dell instrumentation package that enables OptiPlex™, Dell Precision™, and Latitude™ systems to be managed remotely. OMCI contains the underlying driver set that collects system information from a number of different sources on the client computer, including the BIOS, CMOS, System Management BIOS (SMBIOS), System Management Interface (SMI), operating system, APIs, DLLs, and registry settings. OMCI exposes that information through the CIMOM interface of the WMI stack.

Thus, OMCI enables IT administrators to remotely collect asset information, modify CMOS settings, receive proactive notification of potential fault conditions, and be alerted to potential security breaches.

## OMCI Architecture

The OMCI architecture is based on a layered model that is tightly integrated with the Microsoft WMI stack. The top layer consists of management applications such as Dell Client Manager and other standards-based management tools and WMI applications.

These applications request client information via the CIMOM. Beneath the CIMOM is the Dell WMI provider layer, which contains two CIM providers:

- The instance/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 (or events).

These providers register with the CIMOM. When the CIMOM receives a request for information, it routes the request to the appropriate provider. Both Dell and Microsoft providers exist in this layer, and they both provide information on system devices. The data router layer collects information from system components in the data provider layer and passes the information up to the Dell WMI provider layer.

For example, a management console in the WMI application layer requests the available free space on a client system hard drive. The WMI application layer makes the request over the network to the CIMOM (in the Dell WMI provider layer) on the client system. The CIMOM routes the request to the appropriate WMI provider, which, in turn, routes the request through the data router layer to the disk data provider in the data provider layer. The information is then returned (via the same path in reverse) to the management console.

## OMCI Alerts

All alerts that are generated by OMCI are delivered in three ways: Write to the OS Event log, displayed as a dialog box on the local console, and a CIM Indication is generated for the alert. All of the alerts written to the Windows Event Log have a “Source” value of “OMCI”. The alerts supported by OMCI are:

Windows Event Log ID	OMCI Event	OMCI Severity Level
1203	Dell.OMCI.Events.CurrentProbe	Warning
105	Dell.OMCI.Events.DiskCapacity	Warning
1253	Dell.OMCI.Events.ChassisIntrusion	Warning
110	Dell.OMCI.Events.DiskSizeDecreased	Warning
1403	Dell.OMCI.Events.EccMemory	Warning
107	Dell.OMCI.Events.NumberOfProcessorsDecreased	Warning
103	Dell.OMCI.Events.MemorySizeIncreased	Warning
104	Dell.OMCI.Events.MemorySizeDecreased	Warning
106	Dell.OMCI.Events.NumberOfProcessorsIncreased	Warning
102	Dell.OMCI.Events.NumberOfDisksDecreased	Warning
1103	Dell.OMCI.Events.FanProbe	Warning
108	Dell.OMCI.Events.Smart	Warning
101	Dell.OMCI.Events.NumberOfDisksIncreased	Warning
1153	Dell.OMCI.Events.VoltageProbe	Warning
1054	Dell.OMCI.Events.TemperatureProbe	Critical
109	Dell.OMCI.Events.DiskSizeIncreased	Warning

## Sample Scenarios<sup>i</sup>

### Asset Management

A company has several hundred deployed Dell computers. Due to changes in the business and IT staff, inventory information has not been maintained accurately. The CIO requests a plan for identifying which systems can be migrated to Windows Vista. This requires an assessment of the deployed systems to determine the size, scope, and financial impact of such a project.

The man-hours required for manually collecting this information would be substantial. Deploying IT staff to each client system is expensive, not only in IT staff time, but in end-user interruptions.

Using OMCI software on each Dell client system, an IT manager can quickly collect information remotely without dispatching IT staff to each system. Using a tool such as Dell Client Manager or Microsoft Systems Management Server (SMS), the IT manager queries each client system over the network and collects relevant information such as CPU type and speed, amount of memory, hard-drive capacity, BIOS version, and current operating system version. Once collected, the information can be analyzed to determine which systems can be upgraded to Windows Vista.

## Configuration Management

An IT department is planning to standardize the client platform and manage each system through its life cycle. As part of this effort, the department acquired a suite of tools, such as Dell Client Manager, and is planning to automate the deployment of a new client operating system using the Preboot Execution Environment (PXE).

The challenge for the department is to modify the boot order setting in the BIOS of each Dell computer without visiting the desktop. With OMCI installed on each client system, IT has several options for remotely modifying the boot order. Dell Client Manager is a management console that can be used to remotely modify BIOS settings on all Dell OptiPlex™, Dell Precision™, and Latitude™ systems. Another option is to write a VB script that alters the boot order setting (see [“Get/Set Boot Order”](#) in [“Appendix A – Sample Scripts”](#) for an example script on changing the boot order). The script can be delivered remotely over the network and run on each client system.

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 OMCI installed on each client system, the IT department can lock down legacy ports to prevent use of unauthorized peripherals, or enable Wake On LAN (WOL) so the system can be revived from a sleep state during non-peak hours to perform systems management tasks.

## 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, OMCI queries the hard drive with the problem for a predicted failure and passes a SMART alert to the management console. (OMCI also displays the SMART error to the local user.) The alert indicates that the hard drive is experiencing excessive read/write errors. IT contacts the user and recommends an immediate backup of critical data files. A service technician is dispatched with a replacement drive. The hard drive is replaced before it fails, avoiding user downtime, a Help Desk call, and a technician trip to the desktop to diagnose the problem.

## Management Application Support<sup>i</sup>

### Enterprise Consoles

Most leading enterprise management tools and console providers natively support CIM/WMI and do not require custom integration to manage Dell systems instrumented with OMCI. Instead, the management application simply subscribes with the CIMOM, and client-system alerts are forwarded to the management application regardless of the specific client implementation.



## Installing OMCI

OMCI can be installed locally using the OMCI installation wizard, or remotely via a silent or administrative installation.

- **Local Installation** — The OMCI installation wizard allows IT staff to install the complete package or to do a custom installation of specific components required for a particular environment. OMCI can also be modified, repaired, or removed through the Add/Remove Programs dialog in the Windows operating system environment.
- **Silent Installation** — OMCI provides a series of command-line switches for silent installation. This allows IT staff to remotely install the product using an electronic software distribution tool with no end user interaction. To obtain information on performing a silent install, refer to the OMCI User's Guide.
- **Administrative Installation** — The IT Manager can use a server image to deploy OMCI to client computers on a network by using a login script, Windows 2000 system policies, or other methods. The Administrative Installation also provides access directly to the MSI file that can be used to install OMCI. To obtain information on performing an administrative install, refer to the OMCI User's Guide.

## Conclusion

Dell OMCI is available on OptiPlex, Dell Precision, and Latitude client systems. Using OMCI, system administrators can remotely manage assets, monitor system health, and inventory deployed systems in the enterprise.

OMCI interfaces with leading enterprise management consoles that support industry standards. This approach helps to ensure that Dell systems can be managed by a broad array of existing enterprise management tools. OMCI also interfaces with Dell Client Manager.

Finally, tight integration with WMI allows customers to take advantage of the rich scripting capabilities for collecting information and customizing system settings. OpenManage Client Instrumentation is provided at no additional charge to Dell customers.

Additional information and product downloads can be obtained at [www.dell.com/openmanage/](http://www.dell.com/openmanage/). See the DMTF website at [www.dmtf.org](http://www.dmtf.org) for more information on CIM, and the Microsoft development website at [www.microsoft.com](http://www.microsoft.com) for more information on WMI.

## Using OMCI

### Namespace Information

The namespace for access to the Dell OMCI classes is "root\DellOMCI".

## Classes

OMCI provides information in the above mentioned namespace through a number of classes in the namespace. These classes are listed below in general categories. For detailed information on the properties in the classes, refer to the OMCI Reference Guide available at [www.dell.com/openmanage/](http://www.dell.com/openmanage/).

### General System

Dell\_Docking – This class defines general information about the docking station.

Dell\_SystemSummary – This class represents summary information about a computer system.

### Boot devices

Dell\_HardDiskDriveSequence – This class defines the properties used for selecting the priority of hard-disk drives for booting using Dell System Management BIOS.

Dell\_BootDeviceSequence – This class defines the properties used for selecting the boot device order using Dell System Management BIOS.

### BIOS Config

Dell\_Configuration – This class defines properties used for setting parameters in the Dell System Management BIOS.

Dell\_SMBIOSSettings – This class defines properties used for setting parameters in the DellSystem Management BIOS.

Dell\_InternalSpeakerSettings – This class defines the properties for the Click Volume on this portable system.

### HDD

Dell\_SMARTDrive – This class defines the properties for SMART drives.

Dell\_DiskDrive – This class defines the physical properties of disk mass storage devices in the system.

Dell\_LogicalDisk – This class defines properties of the logical drives in a system.

Dell\_DiskPartition – This class represents the capabilities and management capacity of a partitioned area of a physical disk.

### Power

Dell\_PowerManagementSettings – This class defines the properties for power management features on the Notebook.

Dell\_PowerManagementSettingsOnBattery – This class defines the properties for power management features of the Notebook in Battery mode.

### BIOS Flash

Dell\_RemoteFlashBIOSSettings – This class defines settings for Dell Remote BIOS updates.

## Software info

Dell\_SystemManagementSoftware – This class defines the properties of the management software which instruments this component.

## Hardware

Dell\_Battery – This class defines general information about the portable battery(s) in the system.

Dell\_Chassis – This class defines properties of the system enclosure.

Dell\_Fan – This class defines various properties for cooling devices in this system.

Dell\_InfraredController – This class defines properties of infrared controllers.

Dell\_PCCard – This class defines the properties for the notebook PC Cards.

Dell\_ParallelController – This class describes capabilities and management of a parallel port.

Dell\_SCSIController – This class describes capabilities and management of the SCSI controller.

Dell\_SerialController – This class describes capabilities and management of a serial port.

Dell\_USBController – This class describes capabilities and management of a USB controller.

Dell\_VideoController – This class describes capabilities and management of a PC video controller.

Dell\_Processor – This class describes capabilities and management of a processor.

Dell\_Keyboard – This class defines the characteristics of the system keyboard.

Dell\_PointingDevice – This class defines the characteristics of the system pointing device.

Dell\_Slot – This class defines characteristics for different system expansion slots supported by the system.

Dell\_PCIDevice – This class describes capabilities and management of a PCI device controller on an adapter card.

Dell\_PCIBridge – This class describes capabilities and management of a PCI controller providing bridge to bridge capability.

Dell\_PhysicalMemory – This class presents information about individual memory devices in the system. A memory device can be a SIMM, a SIPP, or a single chip, or an entire card-based on the resolution to which a memory error can be determined and whether the device is replaceable as a unit.

Dell\_PhysicalMemoryArray – This class represents an array of physical memory as a collection of one or more Memory Devices associated in sets and partitions.

Dell\_CacheMemory – This class defines properties of System Caches installed in the system.

Dell\_Memory – This class describes capabilities and management of Memory-related logical devices.

Dell\_Monitor – This class describes capabilities and management of a desktop monitor (CRT or LCD).

Dell\_FlatPanel – This class describes Capabilities and management of the flat panel logical device on notebook computers.

Dell\_NetworkAdapter – This class describes characteristics of network adapters in a system.

Dell\_DeviceBay – This class defines device bays characteristics

## Sensors

Dell\_TemperatureSensor – This class describes properties of temperature probes in this system.

Dell\_CurrentSensor – This class describes properties of electrical current probes in this system.

Dell\_VoltageSensor – This class describes properties of voltage probes in this system.

## Other

Dell\_DynamicStates – This class defines the dynamic states information in the portable system.

Dell\_IRQ – This class defines characteristics for IRQs in the system.

Dell\_DMA – This class defines characteristics for DMA channels in the system.

Dell\_Product – This class describes a collection of PhysicalElements, SoftwareFeatures and/or other Products, acquired as a unit.

Dell\_OperatingSystem – An OperatingSystem is software/firmware that makes a ComputerSystem's hardware usable and implements and/or manages the resources, file systems, processes, user interfaces, services, ... available on the ComputerSystem.

Dell\_VideoBIOSElement – VideoBIOSElement represents the low-level software that is loaded into non-volatile storage and used to bring up, configure and access a computer system's videocontroller and display.

Dell\_BIOSElement – BIOSElement represents the low-level software that is loaded into non-volatile storage and used to bring up and configure a computer system.

Dell\_SystemReset – This class defines whether Automatic System Reset functions are available (Capability) and enabled (Status). If the system has a watchdog Timer and the timer is not reset (Timer Reset) before the Interval elapses, an automatic system reset will occur. The system will re-boot according to the Boot Option. This function may repeat until the Limit is reached, at which time the system will re-boot according to the Boot Option at Limit.

Dell\_PortResource – This class presents information about I/O resources.

Dell\_MemoryResource – This class presents information about device memory resources.

## Indications

Dell\_IndicationConfiguration – This class defines the polling configuration for event sources.

Dell\_IndicationStaticValues – This class defines the static values of Dell indications. It has the same properties as class Dell\_Indication, except that it doesn't have the three properties: IndicationTime, Description and PerceivedSeverity. It is used to populate the static values of Dell\_Indication instances.

Dell\_Indication – This class defines the content of Dell indications. It has the same properties as class CIM\_AlertIndication of DMTF Event MOF Specification 2.7 does.

Dell\_IndicationConsumer – This class defines the properties for Dell indication consumer.

## Key Acronyms

- ASF - Alert Standard Format
- CIM - Common Information Model
- CIMOM - CIM Object Manager
- COM/DCOM - Component Object Manager/Distributed Component Object Manager
- DMTF - Distributed Management Task Force
- MOF - Managed Object Format
- OMCI - OpenManage Client Instrumentation
- SNMP - Simple Network Management Protocol
- SMBIOS - System Management BIOS
- SMBus - System Management Bus
- WMI - Windows Management Instrumentation
- WSH - Windows Scripting Host

THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.

## Appendices

### Appendix A – Sample Scripts

#### Clear Chassis Intrusion Status

Below is a sample VB Script that will clear the chassis intrusion status of a system. This script can be run locally or remotely using the name of the system.

Namespace: root\DellOMCI

Class Name: Dell\_SMBIOSsettings

Instance 'Key' Value: 0

Property Name: ChassisIntrusionStatus

Property Value: 5

```
*****
'*** Name: SampleChassisIntrusionClear.vbs
'*** Purpose: To clear the chassis intrusion status on a Dell OMCI client.
'*** Usage: cscript.exe //nologo SampleChassisIntrusionClear.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
*****

Option Explicit

'*** Declare variables
Dim strNameSpace
Dim strComputerName
Dim strClassName
Dim strKeyValue
Dim objInstance
Dim strPropName
Dim strPropValue

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
End If

'*** Initialize variables
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_SMBIOSsettings"
```

```
strKeyValue = "0"
strPropName = "ChassisIntrusionStatus"

'*** Retrieve the instance of Dell_SMBIOSSettings class (there should
'*** only be 1 instance).
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate
,AuthenticationLevel=pktprivacy }//" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))

'*** Set the value of ChassisIntrusionStatus to'5' ("Clear")
'*** Set the new value for the property and save the instance
objInstance.Properties_.Item(strPropName).Value = 5
objInstance.Put_

'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "Clearing Chassis Intrusion Status failed."
End If

'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
    "cscript.exe /nologo SampleChassisIntrusionClear.vbs <systemname>"
WScript.Echo strMessage
End Sub

'*****
```

## Disable Desktop Messages for Fan Alerts

Below is a sample VB Script that will disable the message box that is displayed on the desktop if a fan alert condition is detected. This script may be easily modified to perform an action on any of the other fifteen alerts by changing the 'strKeyValue' value in the script to the desired alert.

Namespace: root\DellOMCI  
Class Name: Dell\_IndicationStaticValues  
Instance 'Key' Value: Dell.OMCI.Events.FanProbe  
Property Name: MaxDisplayNotifications  
Property Value: 0

The list of supported alerts is:

```
Dell.OMCI.Events.CurrentProbe
Dell.OMCI.Events.ChassisIntrusion
Dell.OMCI.Events.DiskCapacity
Dell.OMCI.Events.DiskSizeDecreased
Dell.OMCI.Events.DiskSizeIncreased
Dell.OMCI.Events.EccMemory
Dell.OMCI.Events.FanProbe
Dell.OMCI.Events.MemorySizeDecreased
Dell.OMCI.Events.MemorySizeIncreased
Dell.OMCI.Events.NumberOfDisksDecreased
Dell.OMCI.Events.NumberOfDisksIncreased
Dell.OMCI.Events.NumberOfProcessorsDecreased
```

# Dell OpenManage Client Instrumentation

```
Dell.OMCI.Events.NumberOfProcessorsIncreased
Dell.OMCI.Events.Smart
Dell.OMCI.Events.TemperatureProbe
Dell.OMCI.Events.VoltageProbe
```

```
'*****
'*** Name: SampleDisableFanMessage.vbs
'*** Purpose: To disable the desktop message box for fan alerts on a Dell
'*** OMCI client.
'*** Usage: cscript.exe //nologo SampleDisableFanMessage.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
'*****
```

Option Explicit

```
'*** Declare variables
```

```
Dim strNameSpace
Dim strComputerName
Dim strClassName
Dim strKeyValue
Dim objInstance
Dim strPropName
Dim strPropValue
```

```
'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
```

```
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
End If
```

```
'*** Initialize variables
```

```
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_IndicationStaticValues"
strKeyValue = "Dell.OMCI.Events.FanProbe"
strPropName = "MaxDisplayNotifications"
```

```
'*** Retrieve the instance of Dell_IndicationStaticValues class (there
'*** should be 16 instances - one for each alert type).
```

```
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate," & _
    "AuthenticationLevel=pktprivacy }//" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))
```

```
'*** Set the value of MaxDisplayNotifications to '0'
```

```
'*** Set the new value for the property and save the instance
```

```
objInstance.Properties_.Item(strPropName).Value = 0
objInstance.Put_
```



```
'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "The change to the property failed."
End If

'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
    "cscript.exe /nologo SampleDisableFanMessage.vbs <systemname>"
WScript.Echo strMessage
End Sub

'*****
```

## Disable Desktop Messages for All Alerts

Below is a sample VB Script that will disable the message box that is displayed on the desktop if an alert condition is detected.

Namespace: root\DellOMCI  
Class Name: Dell\_IndicationStaticValues  
Instance 'Key' Value: Not Applicable to this example  
Property Name: MaxDisplayNotifications  
Property Value: 0

```
'*****
'*** Name: SampleDisableAlertMessages.vbs
'*** Purpose: To disable the desktop message box for all alerts on a Dell
'*** OMCI client.
'*** Usage: cscript.exe //nologo SampleDisableAlertMessages.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
'*****
```

Option Explicit

```
'*** Declare variables
Dim objWMIService
Dim strNameSpace
Dim strComputerName
Dim strClassName
Dim ColSystem
Dim objInstance
Dim strPropName
Dim strPropValue

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
```

```
        Call Usage()
        WScript.Quit
    End If

    '*** Initialize variables
    strNameSpace = "root/Dellomci"
    strComputerName = WScript.Arguments(0)
    strClassName = "Dell_IndicationStaticValues"
    strPropName = "MaxDisplayNotifications"

    '*** Establish a connection to the DellOMCI namespace
    Set objWMIService = GetObject("winmgmts:{impersonationLevel=impersonate," & _
        "AuthenticationLevel=pktprivacy}\\\" & strComputerName & "\" & _
        strNameSpace)

    '*** Retrieve the instances of Dell_IndicationStaticValues class (there
    '*** should be 16 instances - one for each alert type).
    Set ColSystem=objWMIService.execquery ("Select * from " & strClassName)

    For each objInstance in ColSystem
        '*** Set the value of MaxDisplayNotifications to'0'
        '*** Set the new value for the property and save the instance
        objInstance.Properties_.Item(strPropName).Value = 0
        objInstance.Put_
    Next

    '*** If any errors occurred, let the user know
    If Err.Number <> 0 Then
        WScript.Echo "The change to the property failed."
    End If

    '*** Sub used to display the correct usage of the script
    Sub Usage()
    Dim strMessage
    strMessage = "incorrect syntax. You should run: " & vbCRLF & _
        "cscript.exe /nologo SampleDisableAlertMessages.vbs <systemname>"
    WScript.Echo strMessage
    End Sub

    '*****
```

## Enable Trusted Platform Module

Below is a sample VB Script that will enable the Trusted Platform Module of a system. This script can be run locally or remotely using the name of the system.

Note: A BIOS Admin password must be set and the password must be specified in the script (replace <Admin\_Password> in the script with the correct BIOS Admin password – leave the quotes around the password). The password will be encrypted by the WMI/DCOM API calls before being sent to the remote system.

Namespace: root\DellOMCI  
Class Name: Dell\_SMBIOSsettings  
Instance 'Key' Value: 0

Property Name: TrustedPlatformModule

Property Value: 3

```
*****
'*** Name: SampleTrustedPlatformModule.vbs
'*** Purpose: To Enable the Trusted Platform Module on a Dell OMCI client.
'*** Usage: cscript.exe //nologo SampleTrustedPlatformModule.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
'***
'*** NOTE: Replace the word password at the end of line 48 (inside the
'*** quotes) with the correct BIOS password if one is set.
*****
```

Option Explicit

```
'*** Declare variables
Dim strNameSpace
Dim strComputerName
Dim strClassName
Dim strKeyValue
Dim objInstance
Dim strPropName
Dim strPropValue

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
End If

'*** Initialize variables
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_Configuration"
strKeyValue = "Configuration"

'*** Retrieve the instance of Dell_Configuration class (there should
'*** only be 1 instance).
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate," & _
    "AuthenticationLevel=pktprivacy}://" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))

'*** Verify the BIOS Admin Password to enable changes to BIOS settings
objInstance.Properties_.Item("Password").Value = "<Admin_Password>"
objInstance.Put_
```

```
'*** Initialize variables
strClassName = "Dell_SMBIOSSettings"
strKeyValue = "0"
strPropName = "TrustedPlatformModule"

'*** Retrieve the instance of Dell_SMBIOSSettings class (there should
'*** only be 1 instance).
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate," &_
    "AuthenticationLevel=pktprivacy}://" &_
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" &_
    Chr(34) & strKeyValue & Chr(34))

'*** Set the value of TrustedPlatformModule to '3' ("Enabled")
'*** Set the new value for the property and save the instance
objInstance.Properties_.Item(strPropName).Value = 3
objInstance.Put_

'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "Enabling Trusted Platform Module failed."
End If

'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
    "cscript.exe /nologo SampleTrustedPlatformModule.vbs <systemname>"
WScript.Echo strMessage
End Sub
'*****
```

## Get/Set Boot Order

Below is a sample VB Script that will get the current Boot Device Sequence of a system. This script can be run locally or remotely using the name of the system.

Namespace: root\DellOMCI

Class Name: Dell\_BootDeviceSequence

```
'*****
'*** Name: SampleGetBootOrder.vbs
'*** Purpose: To get the current Boot Order on a Dell OMCI client.
'*** Usage: cscript.exe //nologo SampleGetBootOrder.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
'*****
```

Option Explicit

```
'*** Declare variables
Dim objWMIService
Dim strNameSpace
```

# Dell OpenManage Client Instrumentation

```
Dim strComputerName
Dim strClassName
Dim ColSystem
Dim objInstance
Dim strPropName
Dim strPropValue

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
End If

'*** Initialize variables
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_BootDeviceSequence"

'*** Establish a connection to the DellOMCI namespace
Set objWMIService = GetObject("winmgmts:{impersonationLevel=impersonate," & _
    "AuthenticationLevel=pktprivacy}\\\" & strComputerName & "\" & _
    strNameSpace)

'*** Retrieve all instances of Dell_BootDeviceSequence
Set ColSystem=objWMIService.execquery ("Select * from " & strClassName)

wscript.echo "Id" & vbTab & "Name" & vbTab & "Order" & vbTab & "Status"

For each objInstance in ColSystem
    '*** Output a tab-delimited list of the current boot order
    wscript.echo objInstance.DellInstanceId & vbTab & _
        objInstance.BootDeviceName & vbTab & _
        objInstance.BootOrder & vbTab & _
        objInstance.Status
Next

'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "An error occurred in obtaining the boot order."
End If

'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
    "cscript.exe /nologo SampleGetBootOrder.vbs <systemname>"
WScript.Echo strMessage
End Sub

'*****
```

A sample tab-delimited output from the preceding script is as follows:

Id	Name	Order	Status
----	------	-------	--------

# Dell OpenManage Client Instrumentation

0	Diskette Drive	1	0	
1	Hard-Disk Drive C:	3	1	
2	IDE CD-ROM Device	2	0	

Extrapolation of the above output shows that the boot order currently is: “Diskette Drive (Order=1), IDE CD-ROM Device(Order=2), Hard-Disk Drive C: (Order=3)” and the “Hard-Disk Drive C:” entry is the only active (“Status” = 1) boot device. The “Order” and “Status” can be changed for each of the items in the list. The “Id” to “Name” mapping will remain constant on a given system. Thus, if the desired boot order/status is:

Id	Name	Order	Status	
0	Diskette Drive	3	1	
1	Hard-Disk Drive C:	2	1	
2	IDE CD-ROM Device	1	1	

Meaning that the boot order is: “IDE CD-ROM Device, Hard-Disk Drive C:, Diskette Drive” and all devices are active, the following example script could be used (there are other ways to make the changes, this is just one). In this example the “Diskette Drive” instance (strKeyValue = “0”) is modified first by changing the “BootOrder” value to 3 and the “Status” value to 1. This will move this boot device to the end of the boot order and will enable it. The output in the above format after this first change would be:

Id	Name	Order	Status	
0	Diskette Drive	3	1	
1	Hard-Disk Drive C:	2	1	
2	IDE CD-ROM Device	1	0	

The values for the “IDE CD-ROM Device” instance (strKeyValue = “2”) are then modified by changing the “BootOrder” value to 1 and the “Status” value to 1 (note in the script below that the only changes to variable required for this are to the strKeyValue and strPropValue1 since the other variables can remain the same values). Note: Changing the “BootOrder” value to 1 in this case is not absolutely necessary since the change to the “Diskette Drive” caused the “IDE CD-ROM Device” to be the first boot device.

```
'*****  
'*** Name: SampleSetBootOrder.vbs  
'*** Purpose: To set the Boot Order on a Dell OMCI client.  
'*** Usage: cscript.exe //nologo SampleSetBootOrder.vbs <systemname>  
'***  
'*** This sample script is provided as an example only, and has not been  
'*** tested, nor is warranted in any way by Dell; Dell disclaims any  
'*** liability in connection therewith. Dell provides no technical  
'*** support with regard to such scripting. For more information on WMI  
'*** scripting, refer to applicable Microsoft documentation.  
'*****
```

Option Explicit

```
'*** Declare variables  
Dim strNameSpace  
Dim strComputerName  
Dim strClassName  
Dim strKeyValue
```

```
Dim objInstance
Dim strPropName1
Dim strPropName2
Dim strPropValue1
Dim strPropValue2

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
End If

'*** Initialize variables
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_BootDeviceSequence"
strKeyValue = "0" '*** DellInstanceId is the "Key" for this class
strPropName1 = "BootOrder"
strPropValue1 = "3"
strPropName2 = "Status"
strPropValue2 = "1"

'*** Retrieve the desired instance of Dell_BootDeviceSequence class
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate
,AuthenticationLevel=pktprivacy }//" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))

'*** Set the two properties and save the instance
objInstance.Properties_.Item(strPropName1).Value = strPropValue1
objInstance.Properties_.Item(strPropName2).Value = strPropValue2
objInstance.Put_

'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "An error occurred in setting the boot order."
End If

'*** Reset values that need to change for the next boot item
strKeyValue = "2"
strPropValue1 = "1"

'*** Retrieve the desired instance of Dell_BootDeviceSequence class
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate
,AuthenticationLevel=pktprivacy }//" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))

'*** Set the two properties and save the instance
objInstance.Properties_.Item(strPropName1).Value = strPropValue1
objInstance.Properties_.Item(strPropName2).Value = strPropValue2
objInstance.Put_

'*** If any errors occurred, let the user know
```

```
If Err.Number <> 0 Then
    WScript.Echo "An error occurred in setting the boot order."
End If
```

```
'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
    "cscript.exe /nologo SampleSetBootOrder.vbs <systemname>"
WScript.Echo strMessage
End Sub
```

```
*****
```

## Change/Set the BIOS Admin Password

Below is a sample VB Script that will get the set or change the BIOS Admin Password. This script can be run locally or remotely using the name of the system.

Note: When changing or setting a BIOS password, there must be a space between the old password and the new password (replace <OldPwd> and <NewPwd> in the script with the desired password values – leave the quotes and the space) even if the old password or new password is blank. The network traffic will be encrypted by the WMI/DCOM API calls before being sent to the remote system.

Namespace: root\DellOMCI  
Class Name: Dell\_Configuration

```
*****
'*** Name: SampleSetAdminPwd.vbs
'*** Purpose: To set the BIOS Admin Password on a Dell OMCI client.
'*** Usage: cscript.exe //nologo SampleSetAdminPwd.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
*****
```

Option Explicit

```
'*** Declare variables
Dim strNameSpace
Dim strComputerName
Dim strClassName
Dim strKeyValue
Dim objInstance

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
```



```
End If

'*** Initialize variables
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_Configuration"
strKeyValue = "Configuration"

'*** Retrieve the instance of Dell_Configuration class (there should
'*** only be 1 instance).
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate," & _
    "AuthenticationLevel=pktprivacy}://" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))

'*** Set the BIOS Admin Password, a space separates the two passwords
'*** in every case even when one of the passwords is blank. The space is
'*** what identifies this as a password change instead of verification.
'*** To set/change the Boot Password, change the property value "Password"
'*** in the next line to "BootPassword".
objInstance.Properties_.Item("Password").Value = "<OldPwd> <NewPwd>"
objInstance.Put_

'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "An error occurred in setting the admin password."
End If

'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
    "cscript.exe /nologo SampleSetAdminPwd.vbs <systemname>"
WScript.Echo strMessage
End Sub
```

```
*****
```

## Change/Set the BIOS Boot Password

Below is a sample VB Script that will get the set or change the BIOS Boot Password. This script can be run locally or remotely using the name of the system.

Note: When changing or setting a BIOS password, there must be a space between the old password and the new password (replace <OldPwd> and <NewPwd> in the script with the desired password values – leave the quotes and the space) even if the old password or new password is blank. The network traffic will be encrypted by the WMI/DCOM API calls before being sent to the remote system.

Namespace: root\DellOMCI  
Class Name: Dell\_Configuration

```
*****
```

```
'*** Name: SampleSetBootPwd.vbs
```

# Dell OpenManage Client Instrumentation

```
'*** Purpose: To set the BIOS Boot Password on a Dell OMCI client.
'*** Usage: cscript.exe //nologo SampleSetBootPwd.vbs <systemname>
'***
'*** This sample script is provided as an example only, and has not been
'*** tested, nor is warranted in any way by Dell; Dell disclaims any
'*** liability in connection therewith. Dell provides no technical
'*** support with regard to such scripting. For more information on WMI
'*** scripting, refer to applicable Microsoft documentation.
'*****
```

Option Explicit

```
'*** Declare variables
Dim strNameSpace
Dim strComputerName
Dim strClassName
Dim strKeyValue
Dim objInstance

'*** Check that the right executable was used to run the script
'*** and that all parameters were passed
If (LCase(Right(WScript.FullName, 11)) = "wscript.exe" ) Or _
    (Wscript.Arguments.Count < 1) Then
    Call Usage()
    WScript.Quit
End If

'*** Initialize variables
strNameSpace = "root/Dellomci"
strComputerName = WScript.Arguments(0)
strClassName = "Dell_Configuration"
strKeyValue = "Configuration"

'*** Retrieve the instance of Dell_Configuration class (there should
'*** only be 1 instance).
Set objInstance = GetObject("WinMgmts:{impersonationLevel=impersonate," & _
    "AuthenticationLevel=pktprivacy}://" & _
    strComputerName & "/" & strNameSpace & ":" & strClassName & "=" & _
    Chr(34) & strKeyValue & Chr(34))

'*** Set the BIOS Boot Password, a space separates the two passwords
'*** in every case even when one of the passwords is blank. The space is
'*** what identifies this as a password change instead of verification.
objInstance.Properties_.Item("BootPassword").Value = "<OldPwd> <NewPwd>"
objInstance.Put_

'*** If any errors occurred, let the user know
If Err.Number <> 0 Then
    WScript.Echo "An error occurred in setting the boot password."
End If

'*** Sub used to display the correct usage of the script
Sub Usage()
Dim strMessage
strMessage = "incorrect syntax. You should run: " & vbCRLF & _
```

# Dell OpenManage Client Instrumentation

---

```
"cscript.exe /nologo SampleSetBootPwd.vbs <systemname>"  
WScript.Echo strMessage  
End Sub
```

```
!*****
```

---

<sup>i</sup> Some information obtained from Dell OpenManage Client Instrumentation (OMCI) Support for the Common Information Model, March 2006