Windows® 7

Portable Command Guide

All the MCTS 70-680, and MCITP 70-685 and 70-686 Commands in One Compact, Portable Resource

PEARSON

DARRIL GIBSON

What Do You Want to Do?

I want to:

Topic	**Chapter**	**Page**
Add drivers to a WinPE image	22	247
Add drivers to offline images	21	240
Add features to offline images	21	240
Append an image to a .wim file	20	233
Capture a Windows 7 image	20	231
Capture user state data	24	265
Check and repair core operating system files	12	135
Check and repair disk integrity	14	172
Check connectivity with computers	13	152
Check local and remote system security	10	116
Configure a proxy server with **netsh**	11	126
Configure a system for remote management	15	184
Configure PowerShell to run scripts	28	309
Convert a disk to NTFS	5	61
Copy permissions when copying files	4	46
Copy text to and from the Command Prompt window	1	9
Create a batch file with Notepad	6	63
Create a bootable VHD file	5	56
Create a hard-link migration store	24	263
Create a list of domain computers	28	315
Create a message box	28	322
Create a system repair disc	14	171
Create and save a PowerShell script in the ISE	27	303
Deploy a Windows 7 image	20	235
Determine whether a file exists	28	318
Disable driver signing checks	14	170
Drag a path from Windows Explorer to the command prompt	3	32
Flush the DNS cache	13	145
Flush the DNS cache with a PowerShell script	28	314
Identify which drivers are signed	12	137
Log an event to an event log	6	73
Manipulate power settings	8	89
Manipulate which processes start on bootup	9	107
Map a drive to a UNC path	4	53

Continues on inside back cover.

Windows 7 Portable Command Guide: MCTS 70-680, and MCITP 70-685 and 70-686

Darril Gibson

800 East 96th Street
Indianapolis, Indiana 46240 USA

Windows 7 Portable Command Guide: MCTS 70-680, and MCITP 70-685 and 70-686

Darril Gibson

ISBN-13: 978-0-7897-4735-8

ISBN-10: 0-7897-4735-9

Library of Congress Cataloging-in-Publication data is on file.

Printed in the United States of America

First Printing: March 2011

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Pearson IT Certification cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

Bulk Sales

Pearson IT Certification offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact

U.S. Corporate and Government Sales
1-800-382-3419
corpsales@pearsontechgroup.com

For sales outside the United States, please contact

International Sales
international@pearson.com

Publisher
Paul Boger

Associate Publisher
Dave Dusthimer

Acquisitions Editor
Betsy Brown

Development Editor
Andrew Cupp

Series Editor
Scott Empson

Managing Editor
Sandra Schroeder

Senior Project Editor
Tonya Simpson

Copy Editor
Bill McManus

Proofreader
The Wordsmithery LLC

Technical Editor
Chris Crayton

Publishing Coordinator
Vanessa Evans

Book Designer
Gary Adair

Composition
Studio Galou, LLC

Contents at a Glance

PART VII: Creating and Managing Images

PART VIII: Working with WinPE and Setup

PART IX: Migrating User Data with USMT

PART X: Using PowerShell

PART XI: Group Policy and the Command Line

Table of Contents

PART II: Working with Files, Folders, and Disks

PART III: Scripting with Batch Files

PART IV: Managing and Maintaining Windows 7

PART V: Troubleshooting Windows 7

PART VI: Remote Administration

PART VII: Creating and Managing Images

PART VIII: Working with WinPE and Setup

PART XI: Group Policy and the Command Line

About the Author

Darril Gibson is the CEO of Security Consulting and Training, LLC. He regularly teaches, writes, and consults on a wide variety of security and technical topics. He's been a Microsoft Certified Trainer for more than ten years and holds several certifications, including MCSE (NT 4.0, 2000, 2003), MCDBA (SQL Server), MCITP (Windows 7, Server 2008, SQL Server), ITIL v3, Security+, and CISSP. He has authored, coauthored, or contributed to more than a dozen books. You can view a listing of most of his current books on Amazon: http://amzn.to/bL0Obo.

Dedication

To my wife, who continues to provide me with love and encouragement. I'm thankful we are sharing our lives together.

Acknowledgments

A book like this is never done in a vacuum. I'm grateful for all the hard work done behind the scenes by the people at Pearson. I'm thankful to Scott Empson, who had the original vision for these books, and grateful that David Dusthimer had faith in me to head up many of the books in the Microsoft series. I especially appreciated the efforts of two key editors, Andrew Cupp and Chris Crayton. This book is much better due to the efforts of these people.

A book like this is never done in a vacuum. I'm grateful for all the hard work done behind the scenes by the people at Pearson. I'm thankful to Scott Empson, who had the original vision for these books, and grateful that David Dusthimer had faith in me to head up many of the books in the Microsoft series. I especially appreciated the efforts of two key editors, Andrew Cupp and Chris Crayton. This book is much better due to the efforts of these people.

About the Series Editor

Scott Empson is the associate chair of the Bachelor of Applied Information Systems Technology degree program at the Northern Alberta Institute of Technology in Edmonton, Alberta, Canada, where he teaches Cisco routing, switching, and network design courses. Scott is also the program coordinator of the Cisco Networking Academy Program at NAIT, a Regional Academy covering Central and Northern Alberta. He has earned three undergraduate degrees: a Bachelor of Arts, with a major in English; a Bachelor of Education, again with a major in English/Language Arts; and a Bachelor of Applied Information Systems Technology, with a major in Network Management. Scott also has a Masters of Education degree from the University of Portland. He holds several industry certifications, including CCNP, CCAI, Network+, and C|EH.

Scott is the series creator and one of the authors of the Portable Command Guide Series. Portable Command Guides are filled with valuable, easy-to-access information to quickly refresh your memory. Each guide is portable enough for use whether you’re in the server room or the equipment closet.

About the Technical Editor

Christopher A. Crayton is an author, technical editor, technical consultant, security consultant, trainer, and SkillsUSA state-level technology competition judge. Formerly, he worked as a computer and networking instructor at Keiser College (2001 Teacher of the Year); as network administrator for Protocol, a global electronic customer relationship management (eCRM) company; and at Eastman Kodak headquarters as a computer and network specialist. Chris has authored several print and online books, including *The A+ Exams Guide*, Second Edition (Cengage Learning, 2008), *Microsoft Windows Vista 70-620 Exam Guide Short Cut* (O’Reilly, 2007), *CompTIA A+ Essentials 220-601 Exam Guide Short Cut* (O’Reilly, 2007), *The A+ Exams Guide*, *The A+ Certification and PC Repair Handbook* (Charles River Media, 2005), *The Security+ Exam Guide* (Charles River Media, 2003), and *A+ Adaptive Exams* (Charles River Media, 2002). He is also co-author of *How to Cheat at Securing Your Network* (Syngress, 2007). As an experienced technical editor, Chris has provided many technical edits/reviews for several major publishing companies, including Pearson Education, McGraw-Hill, Cengage Learning, Wiley, O’Reilly, Syngress, and Apress. He holds MCSE, A+, and Network+ certifications.

We Want to Hear from You!

As the reader of this book, you are our most important critic and commentator. We value your opinion and want to know what we're doing right, what we could do better, what areas you'd like to see us publish in, and any other words of wisdom you're willing to pass our way.

As an associate publisher for Pearson IT Certification, I welcome your comments. You can email or write me directly to let me know what you did or didn't like about this book—as well as what we can do to make our books better.

Please note that I cannot help you with technical problems related to the topic of this book. We do have a User Services group, however, where I will forward specific technical questions related to the book.

When you write, please be sure to include this book's title and author as well as your name, email address, and phone number. I will carefully review your comments and share them with the author and editors who worked on the book.

Email: feedback@pearsonitcertification.com

Mail: David Dusthimer
Associate Publisher
Pearson IT Certification
800 East 96th Street
Indianapolis, IN 46240 USA

Reader Services

Visit our website and register this book at pearsonitcertification.com for convenient access to any updates, downloads, or errata that might be available for this book.

This page intentionally left blank

Introduction

Thanks for buying *Windows 7 Portable Command Guide*. I'd love to say that this book was my idea, but the real credit goes to Scott Empson, who originally developed the vision of this book with Cisco certifications. I've worked with Scott and Pearson Publishing to help bring the same type of books he created for Cisco products to professionals working on Microsoft products. Scott's vision started with the idea that many IT professionals who have already learned the theory still sometimes need help remembering how to implement it.

The book doesn't go into depth teaching these concepts. The idea is that you already understand them. Instead, the goal is to provide enough information to help you remember what you can do and how to do it in a small, portable, and useful journal, not an encyclopedic-sized volume. However, even if a concept is new to you, there's enough information for you to start typing at the command prompt to gain a better understanding.

As an example, you probably know that you can refresh Group Policy from the command prompt, but you might not always remember the exact command is **gpupdate /force**. You might remember that sysprep is used to prepare a computer for imaging, but you might not always remember that the full command is **sysprep /oobe /generalize**. In other words, you know the theory behind why you'd update Group Policy, and why you'd run sysprep, but you might not always remember the syntax. This book is a ready reference of useful commands and procedures with clear-cut examples. It shows the exact syntax of many of the commands needed for administrative tasks performed regularly by Windows 7 administrators.

I started the outline of this book by ensuring that command-prompt commands covered by the Microsoft Certified Information Technology Professional (MCITP) certifications on Windows 7 were included. This includes the 70-680 and 70-685 exams for the MCITP: Enterprise Desktop Support Technician 7 certification, and the 70-680 and 70-686 exams for the MCITP: Enterprise Desktop Administrator 7 certification. I then added the commands I've found valuable in my day-to-day work on networks and from classroom teaching.

Many IT professionals use an engineering journal to help them remember key information needed on the job. It might include specific commands that they sometimes forget, IP addressing schemes used on their networks, steps for important maintenance tasks that are performed infrequently, or anything else they want to easily recall by looking at the journal. If you already have an engineering journal of your own, you can add this as a Windows 7 addendum. If you don't have one, you can start with this book. It includes the same "Create Your Own Journal Here" appendix that Scott uses in the Cisco series. These are blank pages you can use to add your own notes and make this your journal, not mine.

Command Syntax Conventions

The conventions used to present command syntax in this book are as follows:

- **Boldface** indicates syntax that is entered literally as shown.
- *Italic* indicates syntax for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive choices.
- Square brackets ([ ]) indicate an optional element.
- Braces ({ }) indicate a required choice.

CHAPTER 16

Windows Management Instrumentation Command Line

This chapter provides information and commands concerning the following topics:

- Understanding **wmic**
- Configuring the firewall to allow **wmic**
- Running **wmic**
- Modifying the format with the **/format** switch
- Retrieving help from **wmic**
- Understanding aliases
- Using verbs

Understanding wmic

wmic is the command-line implementation of Windows Management Instrumentation (WMI). It extends WMI so that you can execute many WMI commands without a full understanding of the underlying details.

WMI is a group of technologies that allows different applications to interact with the Windows operating system. It is based on the Web-Based Enterprise Management (WBEM) standard and it's a full-blown scripting tool. Administrators use WMI scripting to perform a wide variety of administrative tasks, and WMI scripting is included in many third-party vendor tools.

> **NOTE:** Creating scripts with WMI is beyond the scope of this book. However, the scripting pros at Microsoft have an active website with a lot of rich content on WMI. Check it out here: http://technet.microsoft.com/dd742341.aspx.

Some of the most valuable commands and switches are summarized in the following table.

Command	**Description**

`wmic alias list brief`
`C:\>`**`wmic alias list brief`**
	Retrieves a list of all aliases.

`wmic` *`aliasname`* **`list full`**
`C:\>`**`wmic computersystem  list full`**
	Retrieves a list of all properties and known values for any alias.

`wmic` *`aliasname`* **`get /?`**
`C:\>`**`wmic computersystem  get /?`**
	Retrieves a list of properties for an alias, including the data type and available operations.

`wmic` *`aliasname`* **`set /?`**
`C:\>`**`wmic computersystem  set /?`**
	Retrieves a list of properties for an alias that can be modified. The list includes the data type and available operations.

`/output:`*`target`*
`C:\>`**`wmic /output:c:\data\cpu.txt computersystem list full`**
	Redirects the output to a file. You can redirect the output to the Clipboard by using **/output:clipboard**.

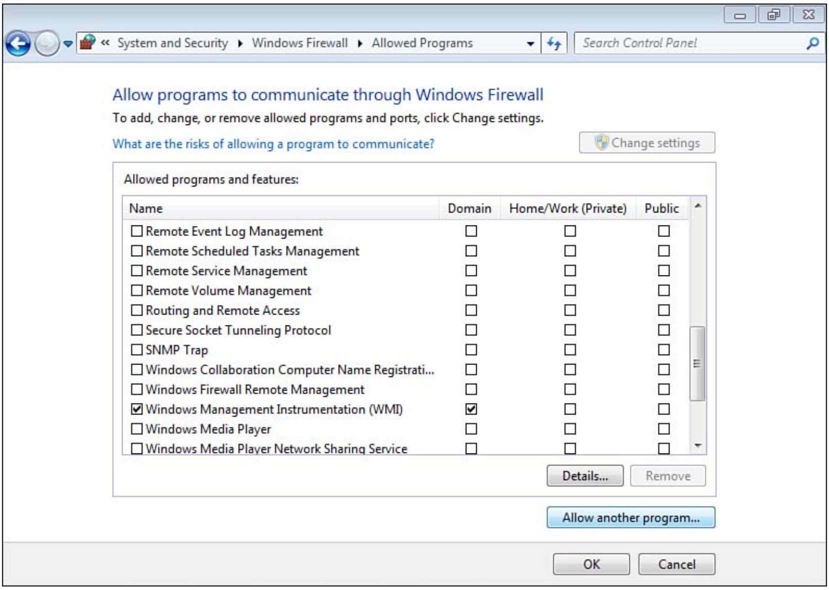
`delete (`*`a process`***`)`**
`wmic process where (name =` *`process-name`***`) delete`**
`C:\>`**`wmic process where (name = 'notepad.exe') delete`**
	Deletes an instance of a running process.

Configuring the Firewall to Allow wmic

If you want to run **wmic** commands on remote computers, you may need to enable the firewall on the remote connections. The primary error you'll see that indicates that wmic commands are prevented by the firewall is "The RPC server is unavailable."

TIP: You'll also see the error "The RPC server is unavailable" if the remote system is unreachable. You can try the **ping** command to determine if the remote system is operational and verify you're using the correct hostname.

You can configure the firewall to allow **wmic** commands by allowing the WMI program through the firewall in the proper profile. Figure 16-1 shows the window for doing so, which you can reach by starting the Control Panel, entering **Firewall** in the Search Control Panel text box, and selecting **Allowing a Program Through Windows Firewall**.



Allow programs to communicate through Windows Firewall

To add, change, or remove allowed programs and ports, click Change settings.

What are the risks of allowing a program to communicate?

Allowed programs and features:

Name	Domain	Home/Work (Private)	Public
□ Remote Event Log Management	□	□	□
□ Remote Scheduled Tasks Management	□	□	□
□ Remote Service Management	□	□	□
□ Remote Volume Management	□	□	□
□ Routing and Remote Access	□	□	□
□ Secure Socket Tunneling Protocol	□	□	□
□ SNMP Trap	□	□	□
□ Windows Collaboration Computer Name Registrati...	□	□	□
□ Windows Firewall Remote Management	□	□	□
☑ Windows Management Instrumentation (WMI)	☑	□	□
□ Windows Media Player	□	□	□
□ Windows Media Player Network Sharing Service	□	□	□

Figure 16-1 Enabling Windows Management Instrumentation in the Firewall

Running wmic

wmic is a shell command similar to **netsh**, covered in Chapter 11, “Configuring Windows 7 with netsh.” You can enter **wmic** from the command prompt to enter the wmic shell. The wmic shell prompt starts in the root\cli name space, from which you can then enter commands. For example, if you want to get detailed information on the computer, you can use the **computersystem list full** command:

```
C:\>wmic
wmic:root\cli>computersystem list full
```

You can also enter the full **wmic** command from the command prompt by preceding it with **wmic**. For example, the following command provides the same output as the previous command:

```
C:\>wmic computersystem list full
```

If you were writing this within WMI (not **wmic**), you would have to understand the query language, and the query would look something like this:

```
Select * from Win32_ComputerSystem
```

However, thanks to the **wmic** built-in aliases, you don’t have to learn the query language to use **wmic**.

The **wmic** command includes several switches. Some of the more common switches are listed in the following table.

TIP: Some commands don't recognize the switch unless it is entered before the command (right after **wmic**) rather than after the command (at the end of the **wmic** command string).

Switch	Description

/? [:brief | :full]
C:\>**wmic /?**
C:\>**wmic /?:full**
	Shows the syntax of all global switches and aliases.
	The default listing is **brief**, but you can also specify **full** to get a more verbose listing of help.

/node:*remotecomputer* **wmic /node:***remotecomputer command* C:\>**wmic /node:win7pcg computersystem** C:\>**wmic /node:win7pcg printer list brief**	You can use the **/node** switch to retrieve information from any remote computers. The first example retrieves information with the computersystem alias, and the second example uses the printer alias.
/user:*username* **wmic /node:***remotecomputer* **/user:***username command* C:\>**wmic /node:win7pcg /user:pearson \administrator computersystem**	Provides the username to be used during the session or for the command. You will be prompted to enter a password. This is useful when connecting to remote systems, but it can't be used to change the credentials on the local system.
/password:*password* **wmic /node:***remotecomputer* **/user:***username* **/password:***password command* C:\>**wmic /node:win7pcg /user:pearson \administrator /password:P@ssw0rd computersystem**	Provides the password to be used with the specified user. This command must be used with a username.

/output:*target* [**stdout** | **clipboard** | *file path and name*]
wmic /**node:***remotecomputer* **/output:***target command*
C:\>**wmic /node:win7pcg /output:clipboard computersystem**
C:\>**wmic /node:win7pcg /output:filename computersystem**
C:\>**wmic /node:win7pcg /output:c:\scripts\test.txt computersystem**
	Identifies where to redirect the output. The output is normally sent to the screen but can be sent to the Clipboard or to a file. The **/output** switch needs to go before the alias.
	When sending it to a file, the path must exist or the command will fail with an "Invalid file name" error.
	TIP: The normal redirect symbol (**>**) can also be used, as in the following example:
	C:\>**wmic /node:win7pcg computersystem > c:\scripts\test.txt**

/append:*target* [**stdout** | **clipboard** | *file path and name*]
C:\>**wmic /node:***remotecomputer* **/append:***filename command*
C:\>**wmic /node:win7pcg /append:c:\scripts\test.txt computersystem**
	Identifies where to redirect the output.
	If the file doesn't exist, it will be created. If it does exist, the output is appended to the file.
	When sending it to a file, the path must exist or the command will fail with an "Invalid file name" error.
	NOTE: The **/append** switch sends the data to the file and to the screen.

Modifying the Format with the /format Switch

The **/format** switch has a few more options that you might find useful. It can be very useful when you combine it with the **/output** switch to send the data to a file in a specific format.

TIP: Each of these commands uses the **computersystem** alias. However, the format of the command is the same with any alias.

Format Switch	Description

`table`
`/format:table`
`C:\>wmic computersystem list full /format:table`
	Formats the output as a table with headers.

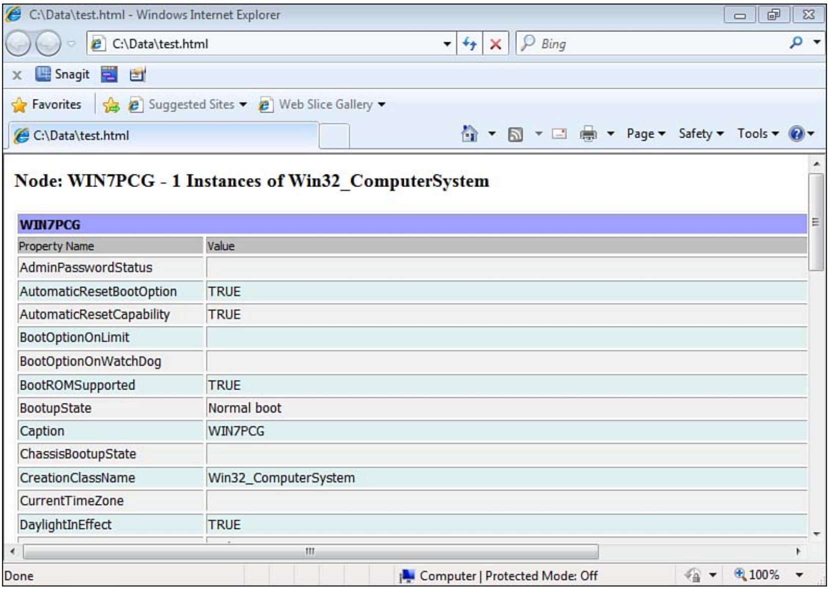
`list`
`/format:list`
`C:\>wmic computersystem list full /format:list`
	Formats the output as a list of each property followed by the value.

`csv`
`/format:csv`
`C:\>wmic computersystem list full /format:csv`
`C:\>wmic /output:c:\data\test.csv computersystem list full /format:csv`
	Formats the output as comma-separated values. The header is displayed first separated by commas. The data is then displayed separated by commas.
	CSV files are easily read in Microsoft Excel, so it's common to use the output switch to send this data to a file.
	NOTE: When you use the output switch, it needs go before the alias.

`xml`
`/format:xml`
`C:\>wmic computersystem list full /format:xml`
`C:\>wmic /output:c:\data\test.xml computersystem list full /format:xml`
	Formats the output in Extensible Markup Language (XML) format.
	If you enter the path to the XML file (such as C:\data\test.xml), the file opens in your web browser.

`hform`
`C:\>wmic computersystem list full /format:hform`
`C:\>wmic /output:c:\data\test.html computersystem list full /format:hform`
	Formats the output as an HTML document. This is useful to display all the properties of an object on a separate row.
	If you enter the path to the HTML file (such as C:\data\test.html), the file opens in your web browser.
	Figure 16-2 shows the output of this command.

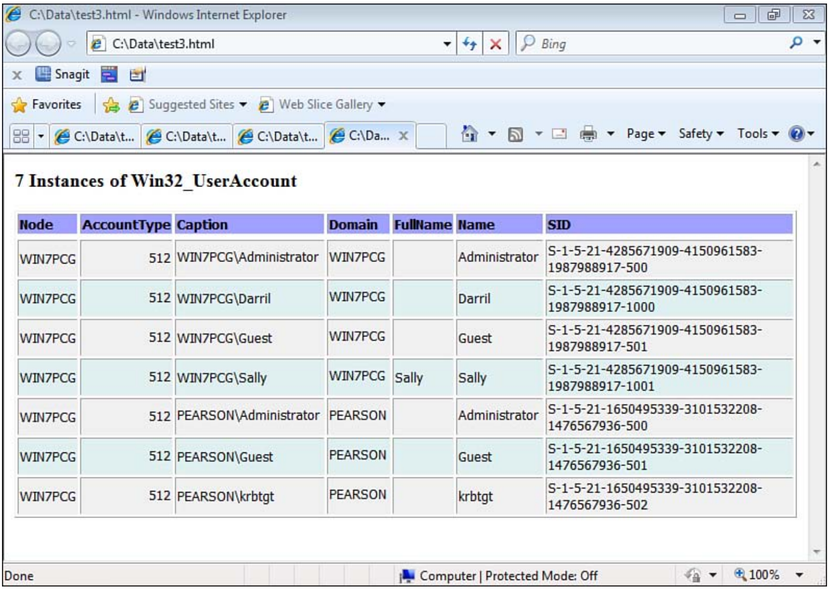
`htable`
`C:\>wmic computersystem list brief /format:htable`
`C:\>wmic /output:c:\data\test3.html useraccount list brief /format:htable`
	Formats the output as an HTML document. In the table format, each object is a single row.
	Figure 16-3 shows the output of this command.



Node: WIN7PCG - 1 Instances of Win32_ComputerSystem

WIN7PCG	
Property Name	Value
AdminPasswordStatus	
AutomaticResetBootOption	TRUE
AutomaticResetCapability	TRUE
BootOptionOnLimit	
BootOptionOnWatchDog	
BootROMSupported	TRUE
BootupState	Normal boot
Caption	WIN7PCG
ChassisBootupState	
CreationClassName	Win32_ComputerSystem
CurrentTimeZone	
DaylightInEffect	TRUE

Figure 16-2 Viewing the Output of **wmic** in Internet Explorer in hform Format



7 Instances of Win32_UserAccount

Node	AccountType	Caption	Domain	FullName	Name	SID
WIN7PCG	512	WIN7PCG\Administrator	WIN7PCG		Administrator	S-1-5-21-4285671909-4150961583-1987988917-500
WIN7PCG	512	WIN7PCG\Darril	WIN7PCG		Darril	S-1-5-21-4285671909-4150961583-1987988917-1000
WIN7PCG	512	WIN7PCG\Guest	WIN7PCG		Guest	S-1-5-21-4285671909-4150961583-1987988917-501
WIN7PCG	512	WIN7PCG\Sally	WIN7PCG	Sally	Sally	S-1-5-21-4285671909-4150961583-1987988917-1001
WIN7PCG	512	PEARSON\Administrator	PEARSON		Administrator	S-1-5-21-1650495339-3101532208-1476567936-500
WIN7PCG	512	PEARSON\Guest	PEARSON		Guest	S-1-5-21-1650495339-3101532208-1476567936-501
WIN7PCG	512	PEARSON\krbtgt	PEARSON		krbtgt	S-1-5-21-1650495339-3101532208-1476567936-502

Figure 16-3 Viewing the Output of **wmic** in Internet Explorer in htable Format

Retrieving Help from wmic

You can retrieve help from **wmic** using multiple methods, as shown in the following table.

> **TIP:** Each of these help commands supports the **/?:full** clause. This sometimes provides more verbose output, but other times it doesn't provide any extra information.

Help Command	Description

/?
C:\>**wmic /?**
C:\>**wmic /?:full**
	Shows the syntax of all global switches and aliases.

switch **/?** */switch_name* **/?** C:\>**wmic /output /?**	Shows information about any single global switch. The example will show help on the output switch.
alias **/?** **wmic** *alias* **/?** C:\>**wmic computersystem /?**	Shows information about aliases in general when the word **alias** is used. If you give the name of an actual alias, it provides information on the alias.

alias verb **/?**
wmic *alias verb* **/?**
C:\>**wmic computersystem get /?**
C:\>**wmic computersystem get /?:full**
C:\>**wmic computersystem set /?**
C:\>**wmic computersystem set /?:full**
	Shows information about one alias and verb combination. This can let you know what properties can be retrieved with the **get** verb and what properties can be configured with the **set** verb.

Understanding Aliases

Aliases are simply friendly names for the detailed query. There are dozens of aliases that you can enter instead of a full **wmic** command. You don't have to understand how the underlying WMI language works to use the alias. For example, the **computersystem** alias can be used to retrieve information on a computer:

```
C:\>wmic computersystem list brief
Domain       Manufacturer           Model            Name
PrimaryOwnerName  TotalPhysicalMemory


Pearson.pub  Microsoft Corporation  Virtual Machine  WIN7PCG
Darril            1610145792

C:\>wmic /node:dc1 computersystem list brief /format:list
```

```
Domain=Pearson.pub
Manufacturer=Microsoft Corporation
Model=Virtual Machine
Name=DC1
PrimaryOwnerName=Windows User
TotalPhysicalMemory=1610063872
```

The **/format:list** switch sends the output as a list instead of a table, which sometimes can be harder to read. The **list brief** clause is used to show some basic details. You can retrieve a much fuller output by using the **list full** clause:

> **TIP:** The **list full** clause sends the output in the list format by default, so this clause is not needed here.

```
C:\>wmic computersystem list full

AdminPasswordStatus=3
AutomaticResetBootOption=TRUE
. . .
Description=AT/AT COMPATIBLE
Domain=Pearson.pub
DomainRole=1
. . .
EnableDaylightSavingsTime=TRUE
. . .
Manufacturer=Microsoft Corporation
Model=Virtual Machine
Name=WIN7PCG
. . .
ThermalState=1
TotalPhysicalMemory=1610145792
UserName=PEARSON\Administrator
WakeUpType=6
Workgroup=
```

> **NOTE:** The entire output for **computersystem list full** spans multiple pages and thus is not listed in its entirety here.

The following tables show many of the aliases that are available. The first column shows the alias friendly name with a short description and its usage. The second column shows the Pwhere usage. If the alias will list multiple items, such as multiple services, you can retrieve data on a single item. WMI uses the Pwhere clause, but with **wmic** you only need to include the name between two single apostrophes. The third column shows the underlying WMI query that is executed.

NOTE: Some items have only a single instance, so a Pwhere clause is not defined within the alias.

Operating System Aliases

The following table shows some aliases that can retrieve data on the operating system.

Alias Friendly Name and Usage	Pwhere Format	WMI Query
computersystem Details on installed operating system and settings `C:\>wmic computersystem list brief`	Not defined	`Select * from Win32_ComputerSystem`
os Operating system details `C:\>wmic os list brief`	Not defined	`Select * from Win32_OperatingSystem`
environment Listing of environment variables `C:\>wmic environment list brief`	Not defined	`Select * from Win32_Environment`
sysdriver Installed services and drivers and current state `C:\>wmic sysdriver list brief`	Where Name='#' `C:\>wmic sysdriver 'disk' list brief`	`Select * from Win32_SystemDriver`
service System services `C:\>wmic service list brief`	Where Name='#' `C:\>wmic service 'winrm' list full`	`Select * from Win32_Service`
process Running processes `C:\>wmic process list brief`	Where ProcessId='#' `C:\>wmic process '6668' list brief`	`Select * from Win32_Process`
startup Identify startup programs `C:\>wmic startup list brief`	Where Caption='#' `C:\>wmic startup 'sidebar' list brief`	`Select * from Win32_StartupCommand`
registry Information on registry `C:\>wmic registry list full`	Not defined	`Select * from Win32_Registry`

`qfe`
Quick fix engineering (hot-fixes)
`C:\>wmic qfe list brief`
	Not defined	`Select * from Win32_QuickFixEngineering`

`nteventlog`
Event logs
`C:\>wmic nteventlog list brief`
	Where LogfileName='#'
	`C:\>wmic nteventlog 'application' list brief`
		`Select * from Win32_NTEventlogFile`

`timezone`
Time zone data
`C:\>wmic timezone list full`
	Not defined	`Select * from Win32_TimeZone`

`bootconfig`
Boot configuration data
`C:\>wmic bootconfig list full`
	Not defined	`Select * from Win32_BootConfiguration`

`recoveros`
Location of recovery OS
`C:\>wmic recoveros list brief`
	Not defined	`Select * from Win32_OSRecoveryConfiguration`

`wmiset`
WMI settings, including whether it's enabled or not
`C:\>wmic wmiset list brief`
	Not defined	`Select * from Win32_WMISetting`

Disk Drive Aliases

These aliases can be used to retrieve information related to disks.

Alias Friendly Name and Usage	Pwhere Format	WMI Query

`diskdrive`
Details on dis3k drive
`C:\>wmic diskdrive list full`
	Where Index='#'
	`C:\>wmic diskdrive  '1' list brief`
		`Select * from Win32_DiskDrive`

`logicaldisk`
Drive data
`C:\>wmic logicaldisk list full`
	Where Name='#'
	`C:\>wmic logicaldisk 'c:' list brief`
		`Select * from Win32_LogicalDisk`

partition Information on disk partitions or volumes C:\>**wmic partition list brief**	Where Index='#' C:\>**wmic partition '0' list full**	**Select * from Win32_ DiskPartition**
diskquota Disk quota settings C:\>**wmic diskquota list full**	Not defined	**Select * from Win32_ DiskQuota**
quotasetting Disk quota settings C:\>**wmic quotasetting list brief**	Not defined	**Select * from Win32_ QuotaSetting**
pagefile Details on paging file(s) C:\>**wmic pagefile list brief**	Not defined	**Select * from Win32_ PageFileUsage**
share Network shares C:\>**wmic** *share* **list brief**	Where Name='#' C:\>**wmic share 'c$' list full**	**Select * from Win32_ Share**
idecontroller IDE disk controllers C:\>**wmic idecontroller list brief**	Not defined	**Select * from Win32_ IDEController**
cdrom CD- and DVD-ROM drives C:\>**wmic cdrom list brief**	Where Drive='#' C:\>**wmic cdrom 'd:' list brief**	**Select * from Win32_ CDROMDrive**

System Hardware Aliases

These aliases can be used to retrieve information on different hardware within the system.

Alias Friendly Name and Usage	Pwhere Format	WMI Query
csproduct Computer system model C:\>**wmic csproduct list full**	Not defined	**Select * from Win32_ ComputerSystemProduct**

cpu Processor information C:\>wmic cpu list full	Where DeviceID='#' C:\>wmic cpu 'cpu0' list brief	Select * from WIN32_PROCESSOR
systemslot Information on expansion slots C:\>wmic systemslot list brief	Not defined	Select * from Win32_SystemSlot
memorychip Memory sticks C:\>wmic memorychip list full	Where Tag = '#' C:\>wmic memorychip ' physical memory 0 ' list brief	Select * from Win32_PhysicalMemory
memphysical Memory totals C:\>wmic memphysical list full	Not defined	Select * from Win32_PhysicalMemoryArray
bios Details on BIOS C:\>wmic bios list full	Not defined	Select * from Win32_BIOS
desktopmonitor Display monitor C:\>wmic desktopmonitor list full	Where DeviceID='#' C:\>wmic desktopmonitor 'desktopmonitor1' list full	Select * from WIN32_DESKTOPMONITOR
nicconfig Configuration of network interface cards (NICs) C:\>wmic nicconfig list brief	Where Index='#' C:\>wmic nicconfig '1' list brief	Select * from Win32_NetworkAdapterConfiguration
nic NICs C:\>wmic nic list brief	Where DeviceID='#' C:\>wmic nic '1' list brief	Select * from Win32_NetworkAdapter
printer Installed printers C:\>wmic printer list brief	Where Name='#' C:\>wmic printer 'Microsoft xps document writer' list full	Select * from Win32_Printer

User, Group, and Domain Aliases

You can use these aliases to get information on objects such as users and groups.

Alias Friendly name and Usage	Pwhere Format	WMI Query
useraccount User account details `C:\>`**`wmic useraccount  list brief`**	Not defined	**`Select * from Win32_UserAccount`**
group User groups `C:\>`**`wmic group list brief`**	Not defined	**`Select * from Win32_Group`**
sysaccount Detailed information on all user and groups, including all the built-in accounts `C:\>`**`wmic sysaccount list brief`**	Where Name='#' `C:\>`**`wmic sysaccount 'everyone' list brief`**	**`Select * from Win32_SystemAccount`**
ntdomain Information on domain (if joined) `C:\>`**`wmic ntdomain list brief`**	Where DomainName='#' `C:\>`**`wmic service 'pearson' list full`**	**`Select * from Win32_NTDomain`**

TIP: These lists of aliases are not complete. If you want to retrieve a full list of all the available aliases, use the command **wmic alias list brief**.

Using Verbs

There are several verbs that can be used with aliases. In simplest terms, the verbs are commands that you can use to work with the aliases.

Verbs (Commands)	Description

`where`
`where (`*`property`***` = "`***`value`***`")`**
`wmic alias where (`*`property`***` = "`***`value`***`") list full`**
`C:\>`**`wmic useraccount where (name = "guest") list full`**
	Use to filter the output. The value must be enclosed in double quotes.
	Only valid properties of the alias can be used in a **where** clause. You can view all valid properties of any alias with the following command:
	`wmic alias list full`

get **get** *property* **wmic /node:***remotecomputer alias* **get** *property* C:\>**wmic /node:win7pcg computersystem get username** C:\>**wmic /node:win7pcg computersystem get username, domain, totalphysicalmemory** C:\>**wmic useraccount where (name = "sally") get** C:\>**wmic useraccount where (name = "sally") get disabled**	You can use the **get** command to retrieve one or more properties of any alias. If you want to retrieve multiple properties, you separate each with a comma. **TIP:** You can identify all the properties you can retrieve from an alias by using the command **wmic alias list full**. You can identify all the properties that can be retrieved with the **wmic alias get /?** command. The first example to the left gets the username of a logged-in user on a remote system of a remote computer. The second example gets the username, the domain, and the amount of physical memory installed on the remote computer. You can also use a **where** clause to filter the data. In the last two examples, a **where** clause is used to retrieve properties on a user account named Sally, and then only the value of the disabled property.
set **set** *property* **=** *"value"* **wmic /node:***remotecomputer alias* **set** *property* **=** *"value"* **wmic /node:***remotecomputer alias* **set** *property* C:\>**wmic /node:win7pcg useraccount where (name = "guest") set disabled = "true"**	The **set** command allows you to set some alias properties. The example combines the **set** command with the **where** clause to disable the guest account on a remote system. The value must be specified in double quotes. **TIP:** You can't set all properties. For example, the memphysical alias reports what physical memory is installed, but you can't change these properties with the **set** command. You can identify all the properties that can be configured with the **wmic alias set /?** command.
delete **wmic** *alias* **where (***property* **=** *value***) delete** C:\>**wmic process where (name = 'notepad.exe') delete**	Deletes an instance. You can use this to terminate processes. The example terminates a running instance of Notepad.

assoc **wmic** *alias* **assoc** C:\>**wmic os assoc** **wmic alias where (***property* **=** '*value*'**) assoc** C:\>**wmic group where (name = 'administrators') assoc**	**assoc** shows the associations with an object. In the first example, it displays information about the operating system alias. The second example shows all WMI objects that are associated with the Administrators group by adding a **where** clause.