

# Windows 7 and Vista Guide to Scripting, Automation, and Command Line Tools



Brian Knittel

**que**<sup>®</sup>

800 East 96th Street  
Indianapolis, IN 46240 USA

# **Windows 7 and Vista: Guide to Scripting, Automation, and Command Line Tools**

**Copyright © 2011 by Pearson Education, Inc.**

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-7897-3728-1

ISBN-10: 0-7897-3728-0

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

Printed in the United States of America

First Printing: December 2010

## **Trademarks**

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Que Publishing 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**

Que Publishing 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 of the United States, please contact

**International Sales**

**international@pearson.com**

**Associate Publisher**  
Greg Wiegand

**Acquisitions Editor**  
Rick Kughen

**Development Editor**  
Todd Brakke

**Managing Editor**  
Sandra Schroeder

**Project Editor**  
Mandie Frank

**Copy Editor**  
Megan Wade

**Indexer**  
Tim Wright

**Proofreader**  
Jovana Shirley

**Technical Editor**  
Ron Barrett

**Publishing Coordinator**  
Cindy Teeters

**Designer**  
Anne Jones

**Compositor**  
Studio Galou, LLC

# E

## Automation Object Reference

This appendix lists the properties and methods of the Automation objects described in this book. For descriptions of the purpose and use of these properties and methods, follow the cross reference to the indicated reference list or table.

### Collection and Dictionary Objects

**JScript Enumerator** enables JScript programs to scan collection objects. See Reference List 3.1 on page 105.

#### **Properties**

- Item
- AtEnd
- MoveFirst
- MoveNext

Note:VBScript has the built-in for each command that loops through collections.

**Scripting.Dictionary** stores and retrieves information in an associative array or symbol table. See Windows Script Host (WSH) documentation.

#### Properties

Count  
CompareMode  
Item  
Key

#### Methods

Add  
Exists  
Items  
Keys  
Remove  
RemoveAll

## Script Management and Utility Objects

**Wscript** provides information about the WSH runtime environment. See Reference List 3.3 on page 120.

#### Properties

Arguments  
BuildVersion  
FullName  
Interactive  
Name  
Path  
ScriptFullName  
ScriptName  
StdErr  
StdIn  
StdOut  
Version

#### Methods

ConnectObject  
CreateObject  
DisconnectObject  
Echo  
GetObject  
Quit  
Sleep

Note:VBScript Err object describes the last error that occurred during the execution of a script. See WSH documentation.

#### Properties

Description  
HelpContext  
HelpFile  
Number  
Source

#### Methods

Clear  
Raise

**Named** describes the set of named arguments (options) passed on the script's command line. See Reference List 9.2 on page 386.

**Properties**

Item  
Length

**Methods**

Count  
Exists

**Unnamed** describes the set of unnamed arguments passed on the script's command line. See Reference List 9.3 on page 389.

**Properties**

Item  
Length

**Method**

Count

**RegExp** searches and parses strings with regular expression patterns. See WSH documentation.

**Properties**

Global  
IgnoreCase  
Pattern

**Methods**

Execute  
Replace  
Test

**Match** describes the results of a RegExp search. See WSH documentation.

**Properties**

FirstIndex  
Length  
Value

## File Access Objects

**Scripting.FileSystemObject** provides access to drives, files, and folders. See Reference List 4.1 on page 125.

**Property**

Drives

**Methods**

BuildPath  
CopyFile  
CopyFolder  
CreateFolder  
CreateTextFile  
DeleteFile  
DeleteFolder  
DriveExists  
FileExists

**Property****Methods**

FolderExists  
 GetAbsolutePathName  
 GetBaseName  
 GetDrive  
 GetDriveName  
 GetExtensionName  
 GetFile  
 GetFileVersion  
 GetFileName  
 GetFolder  
 GetParentFolderName  
 GetSpecialFolder  
 GetStandardStream()  
 GetTempName()  
 MoveFile  
 MoveFolder  
 OpenTextFile

**Drive** describes a single hard drive or drive partition. See Reference List 4.2 on page 136.

**Properties**

AvailableSpace  
 DriveLetter  
 DriveType  
 FileSystem  
 FreeSpace  
 IsReady  
 Path  
 RootFolder  
 SerialNumber  
 ShareName  
 TotalSize  
 VolumeName

**Folder** describes a single folder or directory. See Reference List 4.3 on page 140.

**Properties****Methods**

Attributes  
 DateCreated  
 DateLastAccessed  
 DateLastModified  
 Drive  
 Files

Copy  
 CreateTextFile  
 Delete  
 Move

**Properties**

IsRootFolder  
Name  
ParentFolder  
Path  
ShortName  
ShortPath  
Size  
SubFolders  
Type

**Methods**

**File** describes a single file. See Reference List 4.4 on page 145.

**Properties**

Attributes  
DateCreated  
DateLastAccessed  
DateLastModified  
Drive  
Name  
ParentFolder  
Path  
ShortName  
ShortPath  
Size  
Type

**Methods**

Copy  
Delete  
Move  
OpenAsTextStream

**TextStream** reads from or writes to a text file. See Reference List 4.5 on page 151.

**Properties**

AtEndOfLine  
AtEndOfStream  
Column  
Line

**Methods**

Close  
Read  
ReadAll  
ReadLine  
Skip  
SkipLine  
Write  
WriteBlankLines  
WriteLine

## XML/HTML Processing Objects

**DOMDocument** provides structured access to an XML or HTML file. The following is a full listing. See Reference List 4.6 on page 170 for a discourse on the more important properties.

### Properties

- async
- attributes
- baseName
- childNodes
- dataType
- definition
- doctype
- documentElement
- firstChild
- implementation
- lastChild
- namespaceURI
- nextSibling
- nodeName
- nodeTypedValue
- nodeTypeString
- nodeValue
- ondataavailable
- onreadystatechange
- ontransformnode
- ownerDocument
- parentNode
- parsed
- parseError
- prefix
- preserveWhiteSpace
- previousSibling
- readyState
- resolveExternals
- specified
- text
- url
- validateOnParse
- xml

### Methods

- abort
- appendChild
- cloneNode
- createAttribute
- createCDATASection
- createComment
- createDocumentFragment
- createElement
- createEntityReference
- createNode
- createProcessingInstruction
- createTextNode
- getElementsByTagName
- hasChildNodes
- insertBefore
- load
- loadXML
- nodeFromID
- removeChild
- replaceChild
- save
- selectNodes
- selectSingleNode
- transformNode
- transformNodeToObject



**IXMLDOMNode** represents a single element in an XML or HTML file. See Reference List 4.7 on page 173 for a discourse on more important methods and properties.

### Properties

attributes  
 baseName  
 childNodes  
 dataType  
 definition  
 firstChild  
 lastChild  
 namespaceURI  
 nextSibling  
 nodeName  
 nodeType  
 nodeTypedValue  
 nodeTypeString  
 nodeValue  
 ownerDocument  
 parentNode  
 parsed  
 prefix  
 previousSibling  
 specified  
 text  
 xml

### Methods

appendChild  
 cloneNode  
 hasChildNodes  
 insertBefore  
 removeChild  
 replaceChild  
 selectNodes  
 selectSingleNode  
 transformNode  
 transformNodeToObject

In addition, various node types might have additional properties and methods. For example, when a node represents an XML element (object `IXMLDOMElement`) additional methods such as `getAttribute` and `setAttribute` are available. See Microsoft's online documentation for full details about XML DOM objects.

**IXMLDOMNamedNodeMap** hosts a list of element attributes. The following is a partial listing. See Reference List 4.8 on page 176.

### Property

length

### Methods

getNamedItem  
 getQualifiedItem  
 item()  
 nextNode  
 removeNamedItem  
 removeQualifiedItem  
 reset  
 setNamedItem

## Program Environment Objects

**WScript.Shell** provides an interface to the desktop and current Windows programming environment. See Reference List 4.9 on page 182.

### Properties

CurrentDirectory  
Environment  
SpecialFolders

### Methods

AppActivate  
CreateShortcut  
Exec  
ExpandEnvironmentStrings  
LogEvent  
Popup  
RegDelete  
RegRead  
RegWrite  
Run  
SendKeys

**WshScriptExec** interacts with a program started with the `WScript.Shell` object's `Exec` method. See Reference List 4.10 on page 188.

### Properties

ExitCode  
ProcessID  
Status  
StdErr  
StdIn  
StdOut

### Method

Terminate

**WshShortcut** describes and manages a desktop shortcut. See Reference List 4.11 on page 193.

### Properties

Arguments  
Description  
FullName  
Hotkey  
IconLocation  
RelativePath  
TargetPath  
WindowStyle  
WorkingDirectory

### Method

Save

**WSHEnvironment** describes the current environment variables and manages the initial logon environment definitions. See Reference List 4.12 on page 197.

#### Properties

Item  
Length

#### Methods

Count  
Remove

## Network and Printer Objects

**WSHNetwork** manages a computer's network mappings. See Reference List 5.1 on page 208.

#### Properties

ComputerName  
UserDomain  
UserName

#### Methods

AddPrinterConnection  
AddWindowsPrinterConnection  
EnumNetworkDrives  
EnumPrinterConnections  
MapNetworkDrive  
RemoveNetworkDrive  
RemovePrinterConnection  
SetDefaultPrinter

## Messaging Objects

**CDO.Message** represents an outgoing email message. See Reference List 6.1 on page 236.

#### Properties

Attachments  
AutoGenerateTextBody  
BCC  
BodyPart  
CC  
Configuration  
DataSource  
DSNOptions  
EnvelopeFields  
Fields  
FollowUpTo  
From  
HTMLBody  
HTMLBodyPart  
Keywords  
MDNRRequested  
MIMEFormatted

#### Methods

AddAttachment  
AddRelatedBodyPart  
CreateMHTMLBody  
Forward  
GetInterface  
GetStream  
Post  
PostReply  
Reply  
ReplyAll  
Send

**Properties**

Newsgroups  
 Organization  
 ReceivedTime  
 ReplyTo  
 Sender  
 SentOn  
 Subject  
 TextBody  
 TextBodyPart  
 To

**Methods**

**Fields** holds a set of named parameters for a message or message body part. See Reference List 6.2 on page 243

**Properties**

Count  
 Item

**Method**

Update

**BodyParts** contains a list of message subcomponents. See Reference List 6.3 on page 246.

**Properties**

Count  
 Item

**Methods**

Add  
 Delete  
 DeleteAll

**CD0.BodyPart** contains one message subcomponent, such as an attachment. See Reference List 6.4 on page 247.

**Properties**

BodyParts  
 Charset  
 ContentClass  
 ContentClassName  
 ContentMediaType  
 ContentTransferEncoding  
 DataSource  
 Fields  
 FileName  
 Parent

**Methods**

AddBodyPart  
 GetDecodedContentStream  
 GetEncodedContentStream  
 GetFieldParameter  
 GetInterface  
 GetStream  
 SaveToFile

**CD0.Configuration** configures the delivery method for a message. See Reference List 6.5 on page 251.

Property	Method
Fields	Load

**FAXCOMEx.FaxDocument** represents a transmission to be sent by or received by Windows fax service. See Reference List 6.6 on page 271. The object works differently on XP versus Vista and Windows 7.

Properties	Methods
AttachFaxToReceipt	ConnectedSubmit
Bodies	ConnectedSubmit2
CoverPage	Submit
CoverPageType	Submit2
DocumentName	
GroupBroadcastReceipts	
Note	
Priority	
ReceiptAddress	
ReceiptType	
Recipients	
ScheduleTime	
ScheduleType	
Sender	
Subject	
SubmissionId	

## Windows Management Instrumentation (WMI) Objects

**SWbemLocator** mediates the connections between a client script and WMI service providers on local and remote computers. See “Connecting with the WbemScripting.SWbemLocator Object” on page 291.

Method
ConnectServer

**SWbemServices** represents the connection to a namespace on a managed computer. The following is a partial list. See Reference List 7.1 on page 302.

Property	Methods
Security	Delete
	ExecMethod
	ExecQuery
	InstancesOf

**SWbemObject** provides basic properties for dynamic management objects derived from SWbemObject. See Reference List 7.2 on page 306.

Properties	Methods
Methods	Delete
Path	Instances
Properties	Put

**Win32\_Service** describes a system service process. See Reference List 7.3 on page 315.

Properties	Methods
AcceptPause	Change
AcceptStart	ChangeStartMode
Caption	Create
CheckPoint	Delete
CreationClassName	GetSecurityDescriptor
Description	InterrogateService
DesktopInteract	PauseService
DisplayName	ResumeService
ErrorControl	SetSecurityDescriptor
Name	StartService
PathName	StopService
ProcessId	UserControlService
ServiceSpecificExitCode	
ServiceType	
Started	
StartMode	
StartName	
State	
Status	
SystemCreationClassName	
SystemName	
TagId	
WaitHint	

Note: There are more than 100 other WMI objects.

## Active Directory Scripting Interface Objects

**IADs** provides basic properties for dynamic management objects derived from IADs. See Reference List 8.1 on page 334.

### Properties

AdsPath  
Class  
GUID  
Name  
Parent  
Schema

### Methods

Get  
GetEx  
GetInfo  
GetInfoEx  
Put  
PutEx  
SetInfo

The following IADs objects are derived from IADs and have IADs properties and methods.

**IADsContainer** and **IADsCollection** describe organizational structures and groups. See Reference List 8.2 on page 336.

### Properties

Count  
Filter  
Hints

### Methods

CopyHere  
Create  
Delete  
GetObject  
MoveHere

**IADsComputer** describes one domain member computer. See Reference List 8.3 on page 340.

### Properties

ComputerID  
Department  
Description  
Division  
Location  
MemorySize  
Model  
NetAddresses  
OperatingSystem  
OperatingSystemVersion  
PrimaryUser  
Owner  
Processor  
ProcessorCount

**Properties**

Role  
Site  
StorageCapacity

**IADsDomain** represents a security domain. See Reference List 8.4 on page 343.

**Properties**

AutoUnlockInterval  
Filter  
IsWorkgroup  
LockoutObservationInterval  
MinPasswordAge  
MinPasswordLength  
MaxBadPasswordsAllowed  
MaxPasswordAge  
PasswordAttributes  
PasswordHistoryLength  
SetInfo

**IADsFileService** and **IADsFileServiceOperations** describe and manage the file and printer sharing service on a computer. See Reference List 8.5 on page 345.

**Properties**

Description  
MaxUserCount  
Resources  
Sessions

**IADsResource** describes a single shared resource. See Reference List 8.6 on page 346.

**Properties**

Name  
User  
UserPath  
Path  
LockCount

**IAdsFileShare** describes a single shared folder. See Reference List 8.7 on page 347.

**Properties**

CurrentUserCount  
Description  
HostComputer  
MaxUserCount  
Name  
Path



**IADsGroup** represents a local or domain security group. See Reference List 8.8 on page 349.

#### Properties

Description  
Name

#### Methods

Add  
IsMember  
Members  
Remove

**IADsNameSpaces** lists the ADSI service providers available to the computer running the script. See “IADsNameSpaces” on page 351.

#### Property

DefaultContainer

**IADsPrintJob** describes a queued print job. See Reference List 8.9 on page 352.

#### Properties

Description  
HostPrintQueue  
Notify  
NotifyPath  
Priority  
Size  
StartTime  
TimeSubmitted  
TotalPages  
UntilTime  
User  
UserPath

**IADsPrintJobOperations** manages a print job. See Reference List 8.10 on page 353.

#### Properties

Status  
TimeElapsed  
PagesPrinted  
Position

#### Methods

Pause  
Resume

**IADsPrintQueue** describes the queue for a shared network printer. See Reference List 8.11 on page 354.

#### Properties

BannerPage  
Datatype  
DefaultJobPriority  
Description

**Properties**

HostComputer  
 Location  
 Model  
 Name  
 PrintDevices  
 PrinterPath  
 PrintProcessor  
 Priority  
 Starttime  
 UntilTime

**IAbsPrintQueueOperations** manages a print queue. See Reference List 8.12 on page 356.

**Properties**

PrintJobs  
 Status

**Methods**

Pause  
 Purge  
 Resume

**IAbsService** describes a system service process. See Reference List 8.13 on page 358.

**Properties**

Dependencies  
 DisplayName  
 ErrorControl  
 HostComputer  
 LoadOrderGroup  
 Name  
 Path  
 ServiceAccountName  
 ServiceAccountPath  
 ServiceType  
 StartType  
 StartupParameters  
 Version

**Method**

SetInfo

**IAbsServiceOperations** manages a system service process. See Reference List 8.14 on page 360.

**Property**

Status

**Methods**

Continue  
 Pause  
 SetPassword  
 Start  
 Stop

**IADsSession** describes the connection between a network client and a computer sharing its resources. See Reference List 8.15 on page 361.

### Properties

Computer  
ComputerPath  
ConnectTime  
IdleTime  
User  
UserPath

**IADsUser** describes a domain or local computer user account. See Reference List 8.16 on page 362.

### Properties

AccountDisabled  
Description  
FullName  
Groups  
HomeDirectory  
IsAccountLocked  
LastLogin  
LastLogoff  
Profile

### Methods

ChangePassword  
SetInfo  
SetPassword

**RootDSE** describes the properties of the local Lightweight Directory Access Protocol (LDAP) server. See Reference List 8.17 on page 368.

### Properties

currentTime  
defaultNamingContext  
dnsHostName  
namingContexts  
rootDomainNamingContext  
serverName  
supportedLDAPVersion

**IADsO** and **IADsOU** describe organizational units in an enterprise's business structure. See Reference List 8.18 on page 369.

### Properties

BusinessCategory (IADsOU only)  
Count  
Description  
FaxNumber  
Filter

**Properties**

LocalityName

Name

Parent

PostalAddress

SeeAlso

TelephoneNumber