Using Windows Management Instrumentation Command-line

Table Of Contents

Chapter 1

Chapter 1

Using Windows Management Instrumentation Command-line

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Using Windows Management Instrumentation Command-line

This section covers:

- Running Windows Management Instrumentation Command-line
- Using WMIC help
- WMIC aliases
- WMIC verbs
- WMIC switches
- WMIC commands
- Security with WMIC
- Creating and Customizing WMIC Components

Running Windows Management Instrumentation Command-line

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Running Windows Management Instrumentation Command-line

When you run Windows Management Instrumentation Command-line (WMIC) for the first time, it compiles Alias definitions, or Managed Object Format (MOF) files. Whenever MOF files are changed, WMIC recompiles the files and updates the repository.

You can choose to use WMIC in interactive mode or non-interactive mode. The following gives you some information about the characteristics and functions of the two modes:

Interactive mode

WMIC in interactive mode returns you to the WMIC command prompt after executing a command. **Root\cli** is the default WMIC role, and is there when you starts WMIC in interactive mode. The **Exit** or **Quit** command is used to exit WMIC.

Interactive mode is useful if you are entering a series of WMIC commands.

To start WMIC in interactive mode see Run WMIC in interactive mode.

The following examples include aliases, commands, and global switches that you can enter at the WMIC prompt when in interactive WMIC mode:

- **CONTEXT** - lists all environment variables (Global switches).
- **OS** - displays data about operating system in a default TABLE format.

- **OS list full** - displays **OS** alias data in **LIST** format.
- **/?** - lists descriptions of available environment variables and alias.
  - **OS /?** - lists options available for operating systems.
  - **OS list /?** - specifies formats for "list" operations.
  - **OS list full /?** - lists switches available.
  - **OS list full /format /?** - lists keywords for available formats.
  - **OS call /?** - lists methods (verbs) available.
  - **OS get /?** - lists properties available.

Non-Interactive mode

WMIC in non-interactive mode exits WMIC after performing the command you entered. You are returned to the WMIC command prompt when the command is completed. WMIC aliases, global switches, or commands can also be used in non-interactive mode.

Non-interactive mode is useful if you use WMIC for a batch procedure, or if you only need to execute one WMIC command.

To start WMIC in non-interactive mode see Run WMIC in non-interactive mode.

The following examples show how you can run commands in non-interactive mode:

- <PROMPT>`wmic os get /format:hform>MyOS.htm`

  The output is directed to a html file, and you are returned to the command prompt where you started.

- <PROMPT>`wmic /?`

  The Help is listed, and you are returned to the command prompt where you started

For more information, see Windows Management Instrumentation Command-line.

Using WMIC help

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Using WMIC help

WMIC includes on-line Help at the command line. At any level, you can type **/?** for additional details. By itself, **/?** gives you the global switches and the aliases available in the current role. When used after an alias, **/?** gives you the verbs and switches available for that alias. Following a verb, **/?** gives you the details for that verb.

For example:

For a list of alias and syntax, at the command prompt, type `wmic` to start the WMIC shell and then type:

```
/?
```

To display options available for the Process alias, from within the WMIC shell type:

```
process /?
```

/?:FULL gives all available details. This can be especially useful for the GET and CALL verbs.

Notes

- You can also use a hyphen (**-?**) instead of a forward slash (**/?**).
- For the PATH and CLASS command, progressive help is available for a singular NODE only (on both local and remote computers).

For more information, see Windows Management Instrumentation Command-line.

WMIC aliases

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

WMIC aliases

Aliases are intermediary facilitators that help you access the WMI infrastructure when you are running WMIC. Aliases are friendly names used to capture the features of a WMI class that are relevant to some specific task such as disk or network administration.

Aliases can be used to provide:

- Clearer names for WMI classes, properties, and method.
- Useful output formats of arranged properties.

  The output formats can include specific property values or can be formatted in a manner appropriate to some specific presentation strategy or function. For example, an alias might have a "BRIEF" format that will list only property values essential for the identification of the objects visible through the alias. Management data is retrieved in XML format and processed by built-in or custom XSL output formats. For more information, see Creating and editing formats in WMIC.

WMIC provides aliases for the most commonly used classes of management information. You can add aliases for other classes. Thus, aliases in WMIC may be of the following types:

- Default aliases in the root\cli namespace.
  - For a list of aliases included with WMIC, type `/?` at the WMIC command prompt.
  - For more information about a specific alias, type: `alias_name /?`
  - For more information about alias schema, type: `Alias alias_name list brief`
- Aliases added by you in the root\cli namespace.
- Aliases defined by you in other namespaces. If you define aliases in other namespaces, you must use the **/ROLE** switch to specify the namespace when using it with WMIC. For more information, see Creating and editing aliases.

For information about WMI security, see Managing WMI security.

For information about WMIC security, see Security with WMIC.

For more information, see Windows Management Instrumentation Command-line.

WMIC verbs

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

WMIC Verbs

Most aliases support the following verbs. Aliases might also support other verbs. To use verbs in WMIC, enter the alias name followed by the verb. For more information about **<alias> /?**, see "Aliases" in Help. If an alias does not support a verb, you receive the message "provider is not capable of the attempted operation."

Verb	Action	Parameters or Verb-specific switches	Example
ASSOC	Returns the result of the query: `Associators of {<wmi object>}` Where <wmi object> is the path of objects returned by the **PATH** or **CLASS** commands. The results are instances	Optionally, an output file format, such as LIST, MOF, or other.	OS ASSOC

	associated with the object. When ASSOC is used with an alias, the classes associated with the class underlying the alias are returned. By default, the output for class is in TABLE format. See the following table of switches for /ASSOC.		
CALL	Executes methods.	Method and parameter list if appropriate. Parameter lists are comma delimited. Use **SERVICE CALL /?** to get a list of available methods and their parameters for the current alias.	SERVICE WHERE CAPTION='TELNET' CALL STARTSERVICE
CREATE	Creates a new instance and sets the property values for the new instance. This cannot be used to create a new class.	Properties equated to values, delimited with commas. Use **CREATE /?** for a list of property names for the alias.	ENVIRONMENT CREATE Name="WMIC_test",VariableValue="WMIC_test_value",UserName="*SYSTEM*"
DELETE	Deletes the current instance or set of instances. This can be used to delete a class.	/INTERACTIVE (prompt to confirm) or /NOINTERACTIVE (do not prompt to confirm).	PROCESS WHERE NAME="CALC.EXE" DELETE
GET	Get specific properties.	Property name or switch. See the table of switches for /GET below. Also use **GET /?** for a list of property names and switches for the alias.	PROCESS GET NAME
LIST	Show data. LIST is the default verb.	See the following tables of adverbs and switches for LIST.	PROCESS LIST BRIEF
SET	Property set operations.	Properties equated to values, delimited with commas. Use **SET /?** for a list of property names for the alias.	ENVIRONMENT WHERE Name="WMIC_test" SET VariableValue="WMIC_test_value1"

The **List** verb has the following adverbs. To use adverbs in WMIC, enter the alias name followed by a verb and adverb. For more information about **<alias> /?**, see "Aliases" in Help.

Adverb	**Results**
BRIEF	A core set of the properties.
FULL	The full set of properties. This is the default set of LIST properties.
INSTANCE	The instance paths only.
STATUS	The status and related properties of the object.
SYSTEM	System properties.
Alias-specific or user format	Alias-specific or user defined formats might be defined by providing distinct lists of properties and a format to be used in displaying them.
WRITEABLE	The writeable properties of the objects.

The **List** verb has the following switches. To use verb-specific switches in WMIC, enter the alias name followed by a switch (verbs and adverbs might also be used). For more information about **<alias> /?**, see "Aliases" in Help.

Switch	Effect
/TRANSLATE:<translation table>	Translate the output using the translation table named by the command. BasicXml and NoComma are translation tables included with WMIC.
/EVERY:<interval>	Repeat the command every X seconds; X is determined by the interval.
/FORMAT:<format specifier>	Specify a keyword or XSL file name to format the data, as explained in the following note.

The **Get** verb has the following switches.

Switch	Effect
/VALUE	The output is formatted with each value listed on a separate line and with the name of the property.
/ALL	The output is formatted as a table. The default output format is /ALL.
/TRANSLATE:<translation table>	Translate the output using the translation table named by the command. BasicXml and NoComma are translation tables included with WMIC.
/EVERY:<interval>	Return values every X seconds, X is the interval.
/FORMAT:<format specifier>	Specify a keyword or an XSL file name to format the data, as explained in the following note.

The **Assoc** verb has the following switches.

Switch	Effect
/RESULTCLASS:<classname>	The returned endpoints associated with the source object must belong to or be derived from the specified class.
/RESULTROLE:<rolename>	The returned endpoints must play a particular role in their association with the source object.
/ASSOCCLASS:<assocclass>	The returned endpoints must be associated with the source through the specified class or one of its derived classes.

Notes

- For both the LIST and GET verbs, the format specifier is used to designate either the name of an XSL file or the standard formats TABLE or LIST. **/FORMAT:TABLE** is equivalent to **/ALL**, which is the default output format.
- MOF-formatted output can be captured to a file (using the **/RECORD** switch) and the file can be compiled using Mofcomp.exe on another computer to import it into WMI on the other system. However, make sure that computer-specific properties are edited as appropriate for the other computer.
- The order of the /FORMAT and /TRANSLATE switches is important. If /TRANSLATE follows /FORMAT, the output is formatted and then translated. If /FORMAT follows /TRANSLATE, the output is translated and then formatted.

Examples
The following examples demonstrate the effective use of the verbs and their adverbs and switches.

- PROCESS GET NAME /FORMAT:XML.XSL /TRANSLATE:BASICXML

  The process names are output in XML format. Without the **/TRANSLATE** switch, the characters around the values are displayed as their HTML equivalents.

- PROCESS WHERE Handle="0" ASSOC

  The instances associated with a specific object (unlike the classes associated with an alias) are returned.

For information about WMI security, see Managing WMI security.

For information about WMIC security, authentication, and authorization, see Security with WMIC.

For more information, see Windows Management Instrumentation Command-line.

WMIC switches

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

WMIC switches

Global switches are used to set defaults for the WMIC environment. You can view the current value of the conditions that the global switches set by entering the **CONTEXT** command.

Switch	Condition	Possible values	Default
/NAMESPACE	The namespace that the aliases typically use	Any namespace	root\cimv2
/ROLE	The namespace that WMIC typically looks in for aliases and other WMIC information. When the role changes, the WMIC interactive command prompt changes to match the role. WMIC only has one role by default (\\root\cli), so you only have switch roles if you have a management product that has defined other roles, or if other roles have been created at your organization.	Any namespace that contains aliases.	root\cli
/NODE	Computer names, comma delimited. All commands are synchronously executed against all computers listed in this value. File names must be prefixed with '@'. Computer names within a file must be comma-delimited, or put on separate lines, or both.	Any computer name, a list of computer names, or a file with computer names in it.	Local computer name
/IMPLEVEL	Impersonation level	Anonymous, Identify, Impersonate, Delegate	Impersonate
/AUTHLEVEL	Authentication level	Default, None, Connect, Call, Pkt, Pktintegrity, Pktprivacy	Pktprivacy
/LOCALE	Locale	MS_409 (English), MS_411 (Japanese), MS_40B (Finnish), and so on.	The default language on the computer when WMIC is installed.
/PRIVILEGES	Enable all privileges.	ENABLE or DISABLE	Enabled
/TRACE	The success or failure of all functions used to execute WMIC commands is displayed.	ON or OFF	Off
/RECORD	Records all output to an XML file. Output is also displayed at the command prompt.	File name	Not set and no default file name exists.
/INTERACTIVE	Typically, delete commands are confirmed.	ON or OFF	OFF in NON-INTERACTIVE mode; ON in INTERACTIVE mode
/FAILFAST	Whether or not the **/NODE** computers are checked before trying to execute the WMIC commands against them. When FAILFAST is ON, WMIC pings the computers in the **/NODE** switch before sending WMIC commands to them. If they do not respond to the ping, the WMIC commands are not executed for them.	ON or OFF	OFF
/USER	A user name to be used by WMIC when accessing the /NODE computers or computers specified in aliases. You are prompted for the password. A user name cannot be used with a local computer.	Any user name.	Not set

/PASSWORD	A password to be used by WMIC when accessing the /NODE computers (possibly including the local computer). The password is visible at the command line.	Any password	Not set
/OUTPUT	Specifies a mode for output redirection. All output is directed to the destination given only. Output does not appear at the command line. The destination is cleared before the output begins.	STDOUT, CLIPBOARD, or a file name. STDOUT is the command line. Clipboard is the Windows clipboard. The output can then be pasted to any program that accepts data in the format produced. For more details, see the following note.	STDOUT
/APPEND	Specifies a mode for output redirection. All output is directed to the destination given only. Output does not appear at the command line. The destination is not cleared before the output begins. The new output is appended to the current contents of the destination.	STDOUT, CLIPBOARD, or a file name. STDOUT is the command line. Clipboard is the Windows clipboard. The output can then be pasted to any program that accepts data in the format produced. See the following note for more details.	STDOUT
/AGGREGATE	Used with the **LIST** and **GET/EVERY** switch. If AGGREGATE is ON, **LIST** and **GET** display their results when all computers in the NODE property have either responded or timed out. If AGGREGATE is OFF, **LIST** and **GET** display their results as soon as they are received.	ON or OFF	ON
/AUTHORITY	Specifies the authority type for the connection.	Needed if the value assigned to the IMPLEVEL switch is Delegate (for example, /IMPELEVEL:Delegate). It contains the authority definition string: "kerberos:*TargetDomainName**TargetComputerName*". For this setting to succeed, computers need to have trust for delegation enabled on the Domain Controller.	Not set

Switch	Example
/NAMESPACE	/NAMESPACE:\\root or /NAMESPACE:SMS
/ROLE	/ROLE:\\root
/NODE	/NODE:"*TESTSERVER1*","*TESTSERVER2*",@"*C:\COMPUTERLIST.TXT*"
/IMPLEVEL	/IMPLEVEL:Impersonate
/AUTHLEVEL	/AUTHLEVEL:default
/LOCALE	/LOCALE:MS_411
/PRIVILEGES	/PRIVILEGES:ENABLE
/TRACE	/TRACE:ON
/RECORD	/RECORD:output.xml
/INTERACTIVE	/INTERACTIVE:ON
/FAILFAST	/FAILFAST:ON
/USER	/USER:JSMITH or /USER:"" (to reset the user name and password to nothing).
/PASSWORD	/PASSWORD:PASSWORD
/OUTPUT	/OUTPUT:CLIPBOARD
/APPEND	/APPEND:CLIPBOARD
/AGGREGATE	/AGGREGATE:OFF
/AUTHORITY	/AUTHORITY:"kerberos:*TargetDomainName**TargetComputerName*"

Notes

- To browse for possible ROLE and NAMESPACE namespaces, use the command `WMIC PATH __NAMESPACE`. To see available namespaces in other namespaces, use the **/NAMESPACE** switch to change a target namespace. For example, use `/NAMESPACE:\\root` to change the ROOT namespace.
- The **GET**, **LIST**, and **ASSOC** verbs have WMIC verbs.
- The **/APPEND** and **/OUTPUT** switches can contradict each other (indicating that the results should be sent to different places). When they do contradict each other, **/OUTPUT** predominates in all cases except where **/APPEND** is directing the results to a file. In that case, the results go to both the **/OUTPUT** and **/APPEND** destinations.
- Switches can be used on the same line as commands. For example, you can use `/NODE:"SERVER1" SERVICE` to get a list of all services on SERVER1. The **/NODE** switch works as described above, and all subsequent commands are also applied to SERVER1. However, if **/OUTPUT** and **/APPEND** are used on the same line as a command, the results redirection specified by **/OUTPUT** and **/APPEND** only apply to that command. The results of subsequent commands are directed to the previous value of the OUTPUT and APPEND switches.
- To reset a switch, set its value to "". For example: `/NODE:""` sets NODE back to the current computer name.

For more information, see Windows Management Instrumentation Command-line.

WMIC commands

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

WMIC commands

The following commands are available at all times:

Command	**Purpose**
CLASS	Escape from the default alias mode of WMIC to access classes in the WMI schema directly.
PATH	Escape from the default alias mode of WMIC to access instances in the WMI schema directly.
CONTEXT	Display the current values of all global switches.
QUIT	Exit from WMIC.
EXIT	Exit from WMIC.

For more information about the global switches displayed by CONTEXT, see WMIC switches.

For information about how WMI validates users, see Managing WMI security.

For information about WMIC security, authentication, and authorization, see Security with WMIC.

Creating and Customizing WMIC Components

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Creating and customizing WMIC components

This section covers:

- Alias namespaces and classes.
- Creating and editing aliases.
- Creating and editing formats in WMIC.
- Creating and editing translation tables.

Alias namespaces and classes

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Alias namespaces and classes

Examples of WMI alias classes are listed in the following table:

Alias class	Description
MSFT_CliTranslateTable	Verb-specific switches. WMIC includes translation tables called BasicXml and NoComma.
MSFT_CliTranslateTableEntry	Used in the translation table.
MSFT_CliSeeAlso	The MSFT_CliSeeAlso association provides a relationship between an alias and any aliases with some kind of functional relationship to the alias. You can extend the alias schema to allow aliases to be categorized and navigated in various ways. Use the **SeeAlso** verb of the **Alias** alias to see this information.
MSFT_CliAlias	MSFT_CliAlias class is the principal class in the alias schema. Instances of this class are aliases.

An **MSFT_CliAlias** instance has the following properties:

Property	Description
FriendlyName	The name of the alias. It must be unique.
Description	A description of the alias. This is the descriptive text when **/?** is entered at the WMIC command line.
Formats	A list, each of which has a name and a list of properties (objects of the class MSFT_CliProperty) to be displayed for that format. All formats are objects of the class MSFT_CliFormat.
Verbs	Verbs: A list, each of which are the various behaviors available through this alias. The behaviors come in two forms: • Standard verbs, which are directly supported by the utility. • User-defined verbs, which must map to some method defined for the target of the alias. All verbs are objects of the class MSFT_CliVerb.
Qualifiers	A list, similar to WMI qualifiers. All qualifiers are objects of the class MSFT_Qualifier.
Target	A list, similar to WMI qualifiers. All qualifiers are objects of the class MSFT_Qualifier.
PWhere clause	Optional WHERE clause that limits the Target. It have substitution values which are the parameters of the alias. The substitution values are marked with **#**. If multiple parameters are needed, they are matched with the # markers in sequence.
Connection	Details on which computers to connect to, the security details to be used, and so on. If a connection is not specified, the computers to be accessed is the value of **/NODE**, and the namespace is the value of **/NAMESPACE**. If a user name and password are not provided, then the value of **/USER** and **/PASSWORD**, if available, is used (otherwise the current account is used). Connection is an object of the class MSFT_CliConnection.
View an alias schema	Use a metaalias Alias to view an alias schema; example: `ALIAS OS`

MSFT_CliAlias properties contain the following classes to define objects that are used in the properties:

Class	Used in	Description

MSFT_CliFormat	aliases	The MSFT_CliFormat class defines a named format that determines the set of properties to be displayed by the utility on a **LIST** command. Each MSFT_CliFormat instance has a name, a list of properties and a Format property that contains the name of a style sheet (XSL file) that must be used to format the property display.
MSFT_CliProperty	formats	Each MSFT_CliFormat instance contains a list of MSFT_CliProperty instances. Instances of the MSFT_CliProperty class define the set of properties to be displayed for a given format. Each MSFT_CliProperty instance has a derivation that might be either a constant, the name of a property or a WMI command line **GET** command. Each MSFT_CliProperty also has a localized description. Formats are related to a specific XSL by the NAME property of the format object. FULL, BRIEF, and INSTANCE correspond to particular XSLs, and the NAME can specify the name of a custom XSL file.
MSFT_CliVerb	aliases	Instances of the MSFT_CliVerb class represent verbs defined for the alias. Verbs are either standard verbs (**GET**, **LIST**, **CALL** and **SET**) or user defined. Standard verbs have descriptions and usages but no parameters.
MSFT_CliParam	verbs	
MSFT_CliQualifier	aliases and verbs	WMIC qualifiers serve the same purpose as WMI qualifiers.
MSFT_CliConnection	aliases	Instances of this class provides connection details.

Notes

- When viewing the role namespaces using WMI browsing tools (such as CIM Studio), only the MSFT_CliAlias and MSFT_CliTranslateTable classes have instances. Their properties might include objects of the other classes, but the other classes will not have instances of their own.

- For this reason, when creating aliases (see Creating and editing aliases), you create instances primarily in the MSFT_CliAlias class. You create instances of the other classes as properties of the MSFT_CliAlias class or as properties of its properties.

For more information, see Windows Management Instrumentation Command-line.

Creating and editing aliases

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Creating and editing aliases

You can use the WMI tools to create aliases as Managed Object Format (MOF) files. You can use WMI Common Information Model (CIM) Studio to generate a MOF file for a selected class. This file should be edited to create an instance of MSFT_CliAlias. The MOF files can then be used to register the alias. Aliases are registered with WMI by compiling the MOF form of the alias.

When viewing the role namespaces using WMI browsing tools (such as WMI CIM Studio), only the MSFT_CliAlias and MSFT_CliTranslateTable classes have instances. Their properties might include objects of the other classes, but the other classes do not have instances of their own. For this reason, when creating aliases, you primarily create instances in the MSFT_CliAlias class. You create instances of the other classes as properties of the MSFT_CliAlias class or as properties of its properties.

Localization is supported by using localized text arrays defined in the schema. Localization of aliases descriptions is done through the MFL file (CliEgAliases.mfl).

Adding roles to any WMI namespace can be done, for example, by copying WMIC's CliEgAliases.mof file, and then changing the namespace reference to the namespace that is to be used as a role. The new file can then be compiled with MOFCOMP.exe. Aliases can then be added as described above.

Any WMI namespace can serve as a role; but without the additional steps above, such a role would not have WMIC's interactive help or the base classes required for aliases.

For more information, see Windows Management Instrumentation Command-line.

Creating and editing formats in WMIC

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Creating and editing formats in WMIC

Windows Management Instrumentation Command-line (WMIC) uses two types of formats to display properties: **LIST** format and **OUTPUT** format.

LIST formats allow displaying and recording of the properties of an alias in a specific way, such as:

List Format	Properties
`Process list brief`	displays a short list of properties.
`Process list full`	displays a full list of properties (this is similar to `process get /all`).
`Process list memory`	lists properties related to a process memory.

To see what LIST formats are available for an alias, type:

`<alias name> list /?`.

For example:

```
os list /?
```

Each LIST format has a default output format name specified as part of alias definition. For example, FULL is usually associated with TABLE, while BRIEF, with LIST (value list).

As an optional step, add the LIST formats group to the alias class. This makes the formats discoverable and easier to use. You can also specify the default stylesheet to be used for the format output.

OUTPUT formats are provided by WMIC using the `/FORMAT` switch to display and record output in a desired way. WMIC uses Extensible Style sheet Language (XSL) to transform and format the Extensible Markup Language (XML) output. The provided formats are stored in the WBEM directory as XSL files. Formats are used with **GET**, or **LIST**. Specific formats can be specified using **GET /FORMAT:**, **LIST /FORMAT:**, **GET /ALL**, or **GET /VALUE**.

Use an XSL editor. You might want to first capture some XML data to use for experimenting while using the XSL editor.

Custom formats may be created to display WMIC queries. If you create custom formats, you should also define them in the **Formats** property of the **MSFT_CliAlias** class. Each format must have a name that is unique within the alias.

Other Output formats are included in WMIC. These formats can be set as defaults or as specified using keywords:

Keywords	Output Formats
mof.xsl	Converts XML data to Managed Object Format (MOF) format. If aliases or other WMI objects are defined in XML format, this XSL can be used to convert them to MOFs so that they can be MOFCOMPed. - /FORMAT:MOF format.
texttable.xsl	/ALL or /FORMAT:TABLE.
texttable.xsl	/VALUE format—one line per value.

For more information on mapping, see **XSL-MAPPINGS.XML** in **%windir%\system32\wbem**.

Additional XSL files that are included in WMI and can be useful when you use WMIC are:

XSL Files	
xml.xsl	Similar to rawxml.xsl, except that the output is in HTML format.
hform.xsl	Html output—the data is displayed in a vertical table (one row per property or instance name). The table can be very long.
htable.xsl	Html output—the data is displayed in a horizontal HTML table (one row per instance). The table can be very wide. you can specify the column that the data is typically sorted by. Enclose this file name in quotation marks. The column and its datatype are specified by using the following syntax: `/FORMAT:"htable.xsl":"datatype=<datatype>":"sortby=<property>"`. For example, `PROCESS GET /FORMAT:"htable.xsl":"datatype=number":"sortby=ProcessId"`.
rawxml.xsl	To output XML data in XML format. All WMIC data is extracted in XML format, so this format gives you data the way that WMIC sees it. This can be useful for creating your own XSL files. Outputting data in XML format is also useful if you want to provide the data to systems that accept XML data.

csv.xsl	Comma-delimited—the data is displayed with commas between each value. This format is suitable for importing the data into Microsoft Excel or a similar program.

Caution

- When editing default views, be sure to save your changes to a format outside WMI. If you update WMI or the operating system, the new version of WMIC might return the aliases or output formats to the defaults, overwriting your changes.

For step-by-step procedures on how to create and edit formats, see Create a format in WMIC and Edit a format in WMIC.

For more information, see Windows Management Instrumentation Command-line.

Creating and editing translation tables

Updated: January 21, 2005

Applies To: Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 with SP1, Windows Server 2003 with SP2

Creating and editing translation tables

Like formats, translation tables allow you to change the output of WMIC commands to suit your needs. Translation tables are not as flexible or powerful as formats, but they are much simpler to create and edit.

Translation tables are used when you specify **/TRANSLATE** on the **LIST** or **GET** verbs. WMIC includes translation tables called BasicXML and NoComma. BasicXML converts the HTML symbols for greater-than and less-than characters into the actual characters.

Translation tables are defined in the MSFT_CliTranslateTable WMI class.

For more information, see Windows Management Instrumentation Command-line.