

Using System Variables
WordPerfect Magazine
by Roger Gagon

Don't look now, but WordPerfect is watching. Like Big Brother in George Orwell's *1984*, your WordPerfect program keeps close tabs on many of your WP activities. Most users never know about this information because it can't be accessed without using a macro. Thanks to a macro feature called *system variables*, you can tap into this valuable resource and discover some helpful facts.

This article explains what system variables are and how they're useful in macros. You'll also learn how to create a WP5.1 and WP6.x macro that displays the contents of your system variables (inspired by one of our readers, Floyd Hassinger). This macro is a fabulous tool for troubleshooting problems in WordPerfect, creating advanced macros or just accessing some handy reference information. If you're the office WordPerfect guru, knowing how to use system variables may open your eyes to timesaving macro possibilities you never dreamed of.

WHAT IS A SYSTEM VARIABLE?

A system variable is a piece of information WordPerfect stores in memory. This information describes some aspect of the document on-screen, the current state of WordPerfect or your computer. For example, a system variable might indicate that bold is turned on at the cursor position, that a document is printing or that Reveal Codes or Caps Lock is currently active.

Because these system variables are available as macro commands, you can take advantage of this information in macros you create. For example, you can use the Name system variable to insert the filename of the current document at your cursor position. This was such a common request with 5.1 users that the developers turned this into a feature in 6.x. This is done by pressing Format (Shift+F8), (7) Other, (5) Insert Filename and choosing the desired option.

The system variable commands are a bit different between 5.1 and 6.x. If, for example, you want to insert a document's filename at the cursor position using 5.1, add the command `{SYSTEM}Name~` to your macro. You can also use its numeric equivalent, `{SYSTEM}12~`. To do this in a 6.x macro, use the command `?Name` instead. For example, to type the contents of this system variable, include the command `TYPE(?Name)` in your 6.x macro.

This is a basic use of system variables. They can be used in macros that make decisions or macros that determine a process based on the information provided.

WHAT SYSTEM VARIABLES ARE AVAILABLE?

If you're using 5.1, you can find a detailed list of the system variables in Appendix K of the reference manual under the heading `{SYSTEM}sysvar~`. In case you don't have access to the manual, a summary of that list is included in Figure 1 below.

Attrib (1)	Pos (16)
Cell (2)	Print (17)
CellAttr (23)	Right (18)
CellState (24)	Row (22)
Column (3)	RowState (27)
Document (4)	ShellVer (25)
Endnote (5)	TableBox (19)
Entry (29)	TextBox (20)
Equation (6)	UserBox (21)
Figure (7)	
Footnote (8)	The following system variables
KeyState (26)	were added on later releases of
Left (9)	WordPerfect 5.1:
Line (10)	
List (11)	Version (28)
Menu (13)	Network (30)
Name (12)	Direction (31)
Page (14)	Language (32)
Path (15)	Edit Type (33)

Figure 1. This is a list of the system variables available in WP5.1. To include any of these system variables in a macro, use the command as follows: {SYSTEM}system variable~. You can use either the text or the numeric equivalent as the system variable. For example, if you want to insert a document's filename at the cursor position, add the command {SYSTEM}Name~ or {SYSTEM}12~ to your macro.

If you're using 6.x, you're in luck--you can find a complete list of system variables in the Macro Help screen, assuming you did a full installation of WordPerfect. To access this list, from the pull-down menu bar choose Help, Macros, then cursor down to System Variables Index and press (Enter). You can print this list if you like by pressing Print/Fax (Shift+F7). If you want to see additional help on any of these system variables, highlight the desired system variable and press (L) Look. Press (Esc) when finished to return to the document screen.

As any veteran macro writer knows, these system variables can offer a wealth of information when you're trying to troubleshoot a problem in WordPerfect or create a custom macro application. The problem is, in order to find out what's in any of these system variables, you must first create a makeshift macro just to display the contents of the variable.

If you're using 5.1, you have another hurdle--much of the information returned by system variables is presented in a "masked" value. For example, using Figure 2 (excerpted from the reference manual) as a guide, when bold is the only font attribute turned on, the Attribute system variable ({SYSTEM}Attrib~) equals 4096. When italics is the only font attribute turned on, {SYSTEM}Attrib~ equals 256. When large is the only attribute turned on, {SYSTEM}Attrib~ equals 4. But when all three are turned on, {SYSTEM}Attrib~ equals the sum of all three, or 4356. You have to do the math to break down 4356 into the individual values (4096+256+4), then use Figure 2 to determine that bold, italics and large are all active.

VALUE	DESCRIPTION
0	Normal
1	Extra Large
2	Very Large
4	Large
8	Small
16	Fine
32	Superscript
64	Subscript
128	Outline
256	Italic
512	Shadow
1024	Redline
2048	Double Underline
4096	Bold
8192	Strikethrough
16384	Underline
32768	Small Caps

Figure 2. This list shows the descriptions for each value of the Attrib system variable.

If you'd like to make this information a little more accessible, you can use a macro in 5.1 or 6.x to display the contents of all your system variables from any screen in WordPerfect. Besides displaying the contents of your system variables, the 5.1 macro also breaks down the masked numbers for you and displays only the text messages that apply under each system variable.

Note: You'll want to be able to run this macro from any screen in WordPerfect, so use an Alt-*letter* name. I chose (Alt-S), but you can substitute another letter if you need to.

THE 5.1 MACRO

The 5.1 macro is included as SYSTEM.WPM.

THE 6.X MACRO

First, you'll need to print a list of the system variables (if you didn't do it earlier). From the pull-down menu bar choose Help, Macros, cursor down to System Variables Index and press (Enter). Press Print/Fax (Shift+F7) to print the list.

You can include any or all of these system variables in your macro, but with approximately 350 system variables in 6.x you may not want to type all of them in one sitting. You can start with some of the more common commands and add others later. These common commands are in the included SYSTEM.WPM macro. (This macro is different from the macro of the same name for 5.1.)

You'll want to eliminate about 160 of the system variables that won't do you any good in this macro. These system variables need to be inserted between specific codes in a macro in order to provide any

useful information. Otherwise they simply return a default value of -1. Since this necessary context can't be added to the SYSTEM.WPM macro, you may as well omit these system variables from the list, since they won't be of any use to you. Cross off all of the system variables that begin with the following text: *Border*, *Box* (you can keep *BoxNumber*), *Fill*, *Graphics* and *LineSegment*.

IDEAS ON USING THE 5.1 AND 6.X MACROS

To use the macro, press (Alt-F10), type *system* and press (Enter) anywhere in WordPerfect to display the appropriate screen.

If you're using 6.x, many of these system variables will simply return a value of 0 if the feature is off, or a value of 1 if the feature is on. If you need help interpreting some of the other values, you can highlight any system variable in the 6.x list and choose the Help button to go directly to WP's Help screen for that system variable. Accessing Help ends the macro.

Obviously, the more things you have going on in your document or in WordPerfect, the more status information will be displayed. Here are just a few ideas on how you might want to use each macro:

5.1 and 6.x:

* *Use the macro for troubleshooting problems.* If, for example, you can't figure out why certain font attributes are present in a table cell, even though Reveal Codes shows no attribute codes, run the macro to see which cell attributes are active in a given cell.

* *Place the macro in the network macros directory for end users to run when they call you for support.* Nothing is more frustrating than trying to solve a computer problem over the phone when you can't see the screen. By having a user place the cursor at the location in question, run the macro and read the contents of certain system variables to you over the phone, the cause of a problem may become more evident.

* *Count the number of items in a list.* Have you ever needed to know how many styles you had in a style library? How about the number of fonts you have available in your font list? You can play the macro from within any list (in 6.x, type *list* to cursor to the List system variable) and obtain an accurate count of the current list.

Besides the lists mentioned above, this macro can also quickly obtain the following information:

1. The number of macro commands in the 6.x Macro Commands dialog box. Press (Ctrl+Page Up), (C) Macro Commands.
2. The number of macro commands in the 5.1 Macro Commands list. Press (Home), Macro Define (Ctrl+F10), type a macro name, press (Enter) and (Ctrl-Page Up) to access the list.
3. The number of QuickList entries (press File Manager (F5), (F6) QuickList) and the number of merge codes. From a 6.x merge form file or 5.1 document screen press Merge Codes (Shift+F9) twice.

4. Finally, the macro can obtain the number of search codes (press Search (F2), (F5) Codes) or the number of entries in just about any other feature list.

Note: After running the macro in the 5.1 Macro Commands list, exit back to the document screen to remove the macro prompt.

* *Use the macro when writing other macros.* Using the macro like this is probably something only a macro-writing guru would do. But if you're writing decision-making macros based on the contents of system variables, you need to find out what values are returned under what conditions. Some of these values are listed in Appendices K and T of the 5.1 reference manual and Appendices A and C of the 6.x Macro Help screen, but it's much easier to simply get the value of the code or character at the cursor position by running the macro.

* *Count the number of objects in a long document.* Suppose you had a long document with dozens of graphics, footnotes or endnotes and you needed to find out how many of each the document contained. Go to the bottom of the document, play the macro and note the number next to the appropriate system variable.

6.x only:

* *Document information.* Do you hate popping in and out of Reveal Codes or other dialog boxes to find out what your margins and page size are set to? Run the macro and scroll to the appropriate system variable.

* *Document summary information.* You can examine all of the document summary information for the document currently on-screen by running the macro and scrolling to the *DocSummary* system variables.

You'll be able to use this information in dozens of other ways--these are just a few ideas that I've found useful.