

What's New in the Windows Console in Windows Server 2016 Technical Preview

08/18/2015 • 6 minutes to read

In this article

[Controlling the new features](#)

[Console behavior](#)

[Selection](#)

Applies To: Windows Server Technical Preview

The console host (the underlying code that supports all character-mode applications including the Windows command prompt, the Windows PowerShell prompt, and others) has been updated in several ways to add a variety of new capabilities.

Controlling the new features

The new functionality is enabled by default, but you can switch each of the new features on and off or revert to the previous console host either through the Properties interface (mostly on the **Options** tab) or with these Registry keys (all keys are DWORD values under **HKEY_CURRENT_USER\Console**):

Registry key	Description
ForceV2	1 enables all new console features; 0 disables all new features. Note: this value is not stored in shortcuts, but only in this Registry key.
LineSelection	1 enables line selection; 0 to use block mode only
FilterOnPaste	1 enables new paste behavior
LineWrap	1 wraps text when you resize console windows
CtrlKeyShortcutsDisabled	0 enables new key shortcuts; 1 disables them
ExtendedEdit Keys	1 enables the full set of keyboard selection keys; 0 disables them

Registry key	Description
TrimLeadingZeros	1 trims leading zeroes in selections made by double-clicking; 0 keeps the leading zeros
WindowsAlpha	Sets opacity value between 30% and 100%. Use 0x4C to 0xFF or 76 to 255 to specify value
WordDelimiters	Defines the character that is used to skip to when selecting text a whole word at a time with CTRL+SHIFT+ARROW (the default is the space character). Set this REG_SZ value to contain all characters you want to be treated as delimiters. Note: this value is not stored in shortcuts, but only in this Registry key.

These settings are stored per each window title in the Registry under HKCU\Console. Console windows opened by a shortcut have these settings stored in the shortcut; if the shortcut is copied to another computer, the settings move with it to the new computer. Settings in shortcuts override all other settings, including global settings and defaults. However, if you revert to the original console by using **Use legacy console** in the **Options** tab, this setting is global and will persist for all windows afterwards, including after restarting the computer.

You can pre-configure or script these settings by configuring the Registry appropriately in an Unattend file or with Windows PowerShell.

16-bit NTVDM apps always revert to the older console host.

Note

If you encounter issues with the new console settings and cannot resolve them with any of the specific options listed here, you can always revert back to the original console by setting ForceV2 to 0 or with the **Use legacy console** control in **Options**.

Console behavior

You can now resize the console window at will by grabbing an edge with the mouse and dragging it. Scroll bars only appear if you set window dimensions manually (by using the **Layout** tab in **Properties**) or if the longest line of text in the buffer is wider than the current window size.

The new console window now supports word wrapping. However, if you used console APIs to change text in a buffer, the console will leave the text as it was originally inserted.

Console windows now can be semi-transparent (to a minimum transparency of 30%). You can adjust the transparency from the Properties menu or with these keyboard commands:

To do this:	Use this key combination:
Increase transparency	CTRL+SHIFT+Plus (+) or CTRL+SHIFT+mouse scroll up
Decrease transparency	CTRL+SHIFT+Minus (-) or CTRL+SHIFT+mouse scroll down
Toggle full-screen mode	ALT+ENTER

Selection

There are many new options for selecting text and lines, as well as for marking text and using the buffer history. The console attempts to avoid clashes with applications that might be using the same keys.

For developers: If a conflict occurs, you can typically control the behavior of the application's use of line input, processed input, and echo input modes with the `SetConsoleMode()` API. If you run in processed input mode, the shortcuts below apply, but in other modes, your application must handle them. Any key combinations not listed here function as they did in previous versions of the console. You can also try to resolve conflicts with various settings on the **Options** tab. If all else fails, you can always revert to the original console.

You can now use "click-and-drag" selection outside of QuickEdit mode, and this selection can select text across lines as in Notepad, rather than just a rectangular block. Copy operations no longer require you to remove line breaks. In addition to "click-and-drag" selection, these key combinations are available:

Text selection

To do this:	Use this key combination:
Move the cursor to the left one character, extending the selection	SHIFT+LEFT ARROW

To do this:	Use this key combination:
Move the cursor to the right one character, extending the selection	SHIFT+RIGHT ARROW
Selects text line by line up from the insertion point	SHIFT+UP ARROW
Extends text selection down one line from the insertion point	SHIFT+DOWN ARROW
If the cursor is in the line currently being edited, use this command once to extend the selection to the last character in the input line. Use it a second time to extend the selection to the right margin.	SHIFT+END
If the cursor is not in the line currently being edited, use this command to select all text from the insertion point to the right margin.	SHIFT+END
If the cursor in the line currently being edited, use this command once to extend the selection to the character immediately after the command prompt. Use it a second time to extend the selection to the right margin.	SHIFT+HOME
If the cursor is not in the line currently being edited, use this command to extend the selection to the left margin.	SHIFT+HOME
Extend the selection down one screen	SHIFT+PAGE DOWN
Extend the selection up one screen	SHIFT+PAGE UP
Extend the selection one word to the right. (You can define the delimiters for "word" with the WordDelimiters registry key.)	CTRL+SHIFT+RIGHT ARROW
Extend the selection one word to the left	CTRL+SHIFT+HOME
Extend the selection to the beginning of the screen buffer	CTRL+SHIFT+END
Select all text after the prompt, if the cursor is in the current line and the line is not empty	CTRL+A
Select the entire buffer, if the cursor is not in the current line	CTRL+A

Editing text

You can copy and paste text in the console using keyboard commands. CTRL+C now serves two functions. If no text is selected when you use it, it sends the BREAK command as usual. If text is selected, the first use copies the text and clears the selection; the second use sends BREAK. Here are the other editing commands:

To do this:	Use this key combination:
Paste text into the command line	CTRL+V
Copy selected text to the clipboard	CTRL+INS
Copy selected text to the clipboard; send BREAK	CTRL+C
Paste text into the command line	SHIFT+INS

Mark mode

To enter mark mode at any time, right-click anywhere in the console title bar, point to **Edit**, and select **Mark** from the menu that opens. You can also type CTRL+M. While in mark mode, use the ALT key to identify the start of a line-wrapping selection. (If **Enable line wrapping selection** is disabled, mark mode selects text in a block.) While in mark mode, CTRL+SHIFT+ARROW selects by character and not by word as in normal mode. In addition to the selection keys in the **Editing text** section, these combinations are available in mark mode:

To do this:	Use this key combination:
Enter mark mode to move cursor in the window	CTRL+M
Begin line-wrapping selection in mark mode, in conjunction with other key combinations	ALT
Move cursor in the direction specified	ARROW keys
Move cursor by one page in the direction specified	PAGE keys
Move cursor to beginning of buffer	CTRL+HOME
Move cursor to end of buffer	CTRL+END

Navigating history

To do this:	Use this key combination:
Move up one line in output history	CTRL+UP ARROW
Move down one line in output history	CTRL+DOWN ARROW
Move viewport to top of buffer (if command line is empty) or delete all characters to the left of the cursor (if command line is not empty)	CTRL+HOME
Move viewport to command line (if command line is empty) or delete all characters to the right of the cursor (if command line is not empty)	CTRL+END

Additional keyboard commands

To do this:	Use this key combination:
Open Find dialog	CTRL+F
Close console window	ALT+F4