



<a href="#">Main Page</a>
<a href="#">Blog</a>
<a href="#">Search</a>
<a href="#">FAQ</a>
<a href="#">TOP 10</a>
<a href="#">Pad Files</a>
<a href="#">Contact</a>
<a href="#">About...</a>
<a href="#">Donate</a>
<a href="#">All Utilities</a>
<a href="#">Password Tools</a>
<a href="#">System Tools</a>
<a href="#">Browser Tools</a>
<a href="#">Programmer Tools</a>
<a href="#">Network Tools</a>
<a href="#">Outlook/Office</a>
<a href="#">64-bit Download</a>
<a href="#">Panel</a>
<a href="#">Forensics</a>
<a href="#">Pre-Release Tools</a>
<a href="#">Articles</a>

**NirCmd v2.86**  
**Copyright (c) 2003 - 2019 Nir Sofer**

**Description**

NirCmd is a small command-line utility that allows you to do some useful tasks without displaying any user interface. By running NirCmd with simple command-line option, you can write and delete values and keys in the Registry, write values into INI file, dial to your internet account or connect to a VPN network, restart windows or shut down the computer, create shortcut to a file, change the created/modified date of a file, change your display settings, turn off your monitor, open the door of your CD-ROM drive, and more...

**Examples of what you can do with NirCmd**

Open the door of J: CD-ROM drive	<b>nircmd.exe cdrom open j:</b>
Close the door of Y: CD-ROM drive	<b>nircmd.exe cdrom close y:</b>
Speaks the text currently in the clipboard (For Windows XP/Vista/7/8).	<b>speak text ~\$clipboard\$</b>
Speaks the text stored inside speak.txt into speak.wav filename (For Windows XP/Vista/7/8).	<b>speak file "c:\temp\speak.txt" 0 100 "c:\temp\speak.wav" 48kHz16BitStereo</b>
Increase the system volume by 2000 units (out of 65535)	<b>nircmd.exe changesysvolume 2000</b>
Decrease the system volume by 5000 units (out of 65535)	<b>nircmd.exe changesysvolume -5000</b>
Set the volume to the highest value	<b>nircmd.exe setsysvolume 65535</b>
Mute the system volume	<b>nircmd.exe mutesysvolume 1</b>
Unmute the system volume	<b>nircmd.exe mutesysvolume 0</b>
Switch the system volume between the mute and normal state.	<b>nircmd.exe mutesysvolume 2</b>

Create a shortcut on your desktop that switch the system volume between the mute and normal state.	<b>nircmd.exe cmdshortcut "~\$folder.desktop\$" "Switch Volume" mutesysvolume 2</b>
Turn off the monitor	<b>nircmd.exe monitor async_off</b>
Start the default screen saver	<b>nircmd.exe screensaver</b>
Put your computer in 'standby' mode	<b>nircmd.exe standby</b>
log off the current user	<b>nircmd.exe exitwin logoff</b>
Ask if you want to reboot, and if you answer 'Yes', reboot the computer.	<b>nircmd.exe qboxcom "Do you want to reboot ?" "question" exitwin reboot</b>
Turn off your computer	<b>nircmd.exe exitwin poweroff</b>
Turn off all computers specified in computers.txt !	<b>multiremote copy "c:\temp\computers.txt" exitwin poweroff force</b>
Dial to "My Internet" connection	<b>nircmd.exe rasdial "My Internet"</b>
Disconnect the "My Internet" connection	<b>nircmd.exe rashangup "My Internet"</b>
Make your Internet Explorer windows 75% transparent ! (192 / 256)	<b>nircmd.exe win trans ititle "internet explorer" 192</b>
Minimize all your Internet Explorer windows	<b>nircmd.exe win min class "IEFrame"</b>
Close all your Internet Explorer windows	<b>nircmd.exe win close class "IEFrame"</b>
Close all your Explorer windows (My Computer, folders, and so on)	<b>nircmd.exe win close class "CabinetWClass"</b>
Hide all your Internet Explorer windows	<b>nircmd.exe win hide class "IEFrame"</b>
Show all your Internet Explorer windows (after you made them hidden with previous example)	<b>nircmd.exe win show class "IEFrame"</b>
Center all top-level windows	<b>nircmd.exe win center alltop</b>
Remove the title bar of My Computer window.	<b>nircmd.exe win -style title "my computer" 0x00C00000</b>

Return the title bar of My Computer window that we removed in the previous example.	<b>nircmd.exe win +style title "my computer" 0x00C00000</b>
Set the My Computer window to right-to-left order (For hebrew and arabic languages)	<b>nircmd win +exstyle title "my computer" 0x00400000</b>
Set all child windows of My Computer window to right-to-left order (For hebrew and arabic languages)	<b>nircmd win child title "my computer" +exstyle all 0x00400000</b>
Create a shortcut on your desktop that closes all your Internet Explorer windows	<b>nircmd.exe cmdshortcut " "~\$folder.desktop\$ "Close All IE" win close class "IEFrame"</b>
Create a shortcut on your desktop that hides all your Internet Explorer windows	<b>nircmd.exe cmdshortcut " "~\$folder.desktop\$ "Hide All IE" win hide class "IEFrame"</b>
Create a shortcut on your desktop that shows back all your Internet Explorer windows	<b>nircmd.exe cmdshortcut " "~\$folder.desktop\$ "Show All IE" win show class "IEFrame"</b>
Set the Windows Calculator as top-most window (above all other windows)	<b>nircmd.exe win settopmost title "Calculator" 1</b>
Set the Windows Calculator back to regular window (non top-most window)	<b>nircmd.exe win settopmost title "Calculator" 0</b>
Create a shortcut to Windows calculator under Start Menu->Programs->Calculators	<b>nircmd.exe shortcut "f:\winnt\system32\calc.exe" "~\$folder.programs\$\Calculators" "Windows Calculator"</b>
Hide the desktop window	<b>nircmd.exe win hide class progman</b>
Show the desktop window (After hiding it in previous example)	<b>nircmd.exe win show class progman</b>
Hide the start button on the system tray	<b>nircmd.exe win child class "Shell_TrayWnd" hide class "button"</b>
Show the start button on the system tray	<b>nircmd.exe win child class "Shell_TrayWnd" show class "button"</b>
Hide the clock on the system tray	<b>nircmd.exe win child class "Shell_TrayWnd" hide class "TrayClockWClass"</b>

Show the clock on the system tray	<b>nircmd.exe win child class "Shell_TrayWnd" show class "TrayClockWClass"</b>
Kill (terminate) all instance of Internet Explorer processes	<b>nircmd.exe killprocess iexplore.exe</b>
Create a shortcut on your desktop that opens the door of K: CDROM drive when you run it.	<b>nircmd.exe cmdshortcut "~\$folder.desktop\$" "Open CDROM" cdrom open k:</b>
Create a shortcut to NirSoft Web site on your desktop	<b>nircmd.exe urlshortcut "http://www.nirsoft.net" "~\$folder.desktop\$" "NirSoft"</b>
Add NirSoft Web site to your Favorities under Links folder.	<b>nircmd.exe urlshortcut "http://www.nirsoft.net" "~\$folder.favorites\$\Links" "NirSoft"</b>
Create a shortcut to NirSoft Web site on the desktop of all computers listed in computers.txt	<b>nircmd.exe multiremote copy "c:\temp\computers.txt" urlshortcut "http://www.nirsoft.net" "~\$folder.common_desktop\$" "NirSoft"</b>
Set the display mode to 800x600x24bit colors	<b>nircmd.exe setdisplay 800 600 24</b>
Create a shortcut on the desktop that set the display mode to 800x600x24bit colors	<b>nircmd.exe cmdshortcut "~\$folder.desktop\$" "800x600x24" setdisplay 800 600 24</b>
Copy all shortcuts on your desktop to another folder (f:\temp\desktop).	<b>nircmd.exe execmd copy "~\$folder.desktop\$\.lnk" f:\temp\desktop</b>
Restart your Apache server (under Windows NT/2000/XP/2003)	<b>nircmd.exe service restart apache</b>
Create a shortcut on your desktop that restarts the Apache server	<b>nircmd.exe cmdshortcut "~\$folder.desktop\$" "Restart Apache" service restart apache</b>
Restart your IIS	<b>nircmd.exe service restart w3svc</b>
Restart MySQL	<b>nircmd.exe service restart MySQL</b>
Open the desired Registry key/value in RegEdit	<b>nircmd.exe regedit "HKLM\Software\Microsoft\Windows\CurrentVersion" "CommonFilesDir"</b>
Open the Registry key that you copied to the clipboard in RegEdit.	<b>nircmd regedit "~\$clipboard\$"</b>
Disable the screen saver	<b>nircmd.exe regsetval sz "HKCU\control panel\desktop" "ScreenSaveActive" 0</b>

Enable the screen saver	<b>nircmd.exe regsetval sz "HKCU\control panel\desktop" "ScreenSaveActive" 1</b>
Change the date/time of the specified filename (creation time and modified time)	<b>nircmd.exe setfiletime "c:\temp\myfile.txt" "24-06-2003 17:57:11" "22-11-2005 10:21:56"</b>
Copy your desktop folder path to the clipboard	<b>nircmd.exe clipboard set ~\$folder.desktop\$</b>
Copy your start menu folder path to the clipboard	<b>nircmd.exe clipboard set ~\$folder.start_menu\$</b>
Copy the content of info1.txt (simple text file) to the clipboard	<b>nircmd.exe clipboard readfile "c:\My Files\info1.txt"</b>
Add the text content of clipboard to info1.txt	<b>nircmd.exe clipboard addfile "c:\My Files\info1.txt"</b>
Clear the clipboard	<b>nircmd.exe clipboard clear</b>
Create all folders specified in "c:\temp\folders.txt". The folder path names are separated by CRLF characters.	<b>nircmd.exe paramsfile "c:\temp\folders.txt" "" "" execmd md ~\$param.1\$</b>
Install the specified .NET assembly in the global assembly cache (like gacutil)	<b>nircmd.exe gac install "C:\temp\MyAssembly\bin\MyAssembly.dll"</b>
Empty the recycle bin in all drives.	<b>nircmd.exe emptybin</b>
Answer 'Yes' to a standard Windows message-box.	<b>nircmd.exe dlg "" "" click yes</b>
Wait 2 seconds, and then save the current screen to shot.png	<b>nircmd.exe cmdwait 2000 savescreenshot "f:\temp\shot.png"</b>
Save 10 screenshots in a loop, and wait 60 seconds between the screenshot save calls. The filenames of the screenshot will contain the time and date of the saved screenshot.	<b>nircmd.exe loop 10 60000 savescreenshot c:\temp\scr~\$currdate.MM_dd_yyyy\$~\$curtime.HH_mm_ss\$.png</b>
Wait until Firefox is closed, and then say "Firefox was closed"	<b>nircmd.exe waitprocess firefox.exe speak text "Firefox was closed"</b>
Run RegEdit with SYSTEM user (On Windows 7/8/Vista/2008), which allows you to view all hidden system Registry keys	<b>nircmd.exe elevatecmd runassystem c:\windows\regedit.exe</b>

## System Requirements

This utility can work in all versions of Windows operating system: Windows 9x/ME, Windows NT, Windows 2000, Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, Windows 8, and Windows 10.

## Versions History

Date	Version	Description
01/08/2019	2.86	<ul style="list-style-type: none"> <li>Fixed bug: elevate and other commands failed to work properly when there were many parameters after the command.</li> </ul>
23/06/2019	2.85	<ul style="list-style-type: none"> <li>The runinteractive and runinteractivecmd commands now work properly even if you don't run them from a Windows service.</li> <li>Fixed bug: The behaviour of consolewrite command was unexpected when there was '%' character.</li> </ul>
22/04/2019	2.84	<ul style="list-style-type: none"> <li>Fixed to work again on Windows 98/2000.</li> </ul>
19/04/2019	2.83	<ul style="list-style-type: none"> <li>Added runinteractive and runinteractivecmd commands, which allows you to run a process or NirCmd command that accesses the GUI from a Windows service.</li> <li>For example, if you try to use the savescreenshot command from a Windows service, you'll get a black image, because NirCmd cannot access the user interface from a Windows service.</li> <li>You can solve this problem with runinteractivecmd command: nircmd.exe runinteractivecmd savescreenshot "c:\temp\screenshot1.png"</li> <li>Be aware that runinteractive and runinteractivecmd commands only work from a Windows service ('SYSTEM' user).</li> </ul>
22/03/2019	2.82	<ul style="list-style-type: none"> <li>Fixed the savescreenshot or savescreenshotfull commands to work properly with DPI scaling.</li> </ul>
23/05/2016	2.81	<ul style="list-style-type: none"> <li>Fixed bug: NirCmd failed to execute the cmdwait, loop and a few other commands after qboxcom and qboxcomtop commands.</li> </ul>
30/01/2016	2.80	<ul style="list-style-type: none"> <li>Fixed bug: clipboard readfile failed to read the last character of Ascii file.</li> <li>Added writeufile and adddufile to clipboard command, for writing files in Unicode.</li> <li>Fixed the restartexplorer command to work properly on Windows 10.</li> </ul>
10/08/2013	2.75	<ul style="list-style-type: none"> <li>Added new commands: moverecyclebin, suspendprocess, resumeprocess.</li> <li>Added etitle option to the find parameter of win command.</li> <li>'clipboard readfile' command now reads Unicode files (UTF-8 and UTF-16).</li> </ul>

29/10/2012	2.71	<ul style="list-style-type: none"> <li>Added runassystem command, which allows you to run a process with SYSTEM user (Works on Windows 7/8/Vista/2008)</li> </ul>
12/09/2012	2.70	<ul style="list-style-type: none"> <li>Added new option to speak command: You can now save the entire speech into a .wav file that can be played with any media player (Instead of playing it directly to your sound card).</li> <li>Added restartexplorer command, which simply restarts Windows Explorer gracefully.</li> <li>Added screensavertimeout command, which allows you to set the screensaver timeout, in seconds.</li> <li>Added togglemax, togglemin, togglehide, toggledisable actions to win command.</li> <li>Added SystemSounds support for setappvolume, changeappvolume, and muteappvolume commands.</li> <li>Fixed issue on Windows 7: 'win close alltopnodesktop' command caused a shutdown dialog-box to appear.</li> <li>Fixed the setprimarydisplay to work properly.</li> </ul>
28/06/2011	2.65	<ul style="list-style-type: none"> <li>Added setdefaultsounddevice command (for Windows 7/Vista/2008 only), which allows you to set the default sound device.</li> <li>Added setsubunitvolumedb command (for Windows 7/Vista/2008 only), which allows you to set the volume of sound device subunits (Line In, Microphone, and so on)</li> <li>Added mutesubunitvolume command (for Windows 7/Vista/2008 only), which allows you to mute/unmute the volume of sound device subunits (Line In, Microphone, and so on)</li> </ul>
21/06/2011	2.60	<ul style="list-style-type: none"> <li>Added mediaplay command for playing audio file (.mp3, .wav, and so on) for the specified number of milliseconds.</li> <li>Added changebrightness and setbrightness commands for setting the screen brightness on laptops and netbook computers.</li> </ul>
08/06/2011	2.55	<ul style="list-style-type: none"> <li>On savescreenshot, savescreenshotfull, and savescreenshotwin commands, you can now specify *clipboard* as a filename, in order to save the screenshot into the clipboard</li> <li>On savescreenshot command, you can now specify a rectangle to capture as x, y, width, and height instead of capturing the entire screen.</li> <li>Fixed savescreenshotfull command to work properly on multiple screen system with negative coordinates.</li> </ul>
27/03/2011	2.52	<ul style="list-style-type: none"> <li>Added 'runas' command, which allows you to run a new process with the specified user name/password.</li> <li>Added 'Flash' action to the win command, which allows you to flash the specified window.</li> </ul>
13/03/2011	2.51	<ul style="list-style-type: none"> <li>Added 'focused' option for setappvolume, changeappvolume, and muteappvolume commands, which allow you to easily change the volume of the application that is currently in focus, on Windows 7/Vista/2008.</li> </ul>

		<ul style="list-style-type: none"> <li>Added 'default_record' option for setsysvolume, changesysvolume, setsysvolume2, changesysvolume2, and mutesysvolume commands, which allows you set easily change the volume of default recording device, on Windows 7/Vista/2008.</li> </ul>
24/02/2011	2.50	<ul style="list-style-type: none"> <li>Added 3 new sound volume commands to change or mute application volume on Windows 7/Vista/2008: setappvolume, changeappvolume, muteappvolume</li> <li>Added new information command that shows you a message box with all sound devices on your Windows 7/Vista/2008 system: showsounddevices.</li> </ul>
22/02/2011	2.48	<ul style="list-style-type: none"> <li>Fixed the problems with the timeout of the tray balloon in trayballoon command.</li> </ul>
20/01/2011	2.47	<ul style="list-style-type: none"> <li>Fixed bug: The sendkeypress command didn't work properly with Fxx keys (F1, F2, and so on).</li> </ul>
28/11/2010	2.46	<ul style="list-style-type: none"> <li>Added option to locate window by process in win command.</li> <li>Added max and min options (Minimized and maximized windows) into exec and exec2 commands.</li> <li>Added x64 builds of nircmd.exe and nircmdc.exe</li> </ul>
18/10/2010	2.45	<ul style="list-style-type: none"> <li>Added trayballoon command - displays a tray balloon with your desired text and icon.</li> <li>Added sendkeypress command - Alternative to the sendkey command. Allows you to send key press combination in much easier way, for example: sendkeypress ctrl+shift+esc</li> <li>Added clonefiletime command to clone the date/time of existing file into one or more files.</li> <li>Added setconsolemode command to set the console display mode - window or full screen.</li> <li>Added setconsolecolor command to set the foreground and background colors of the console text.</li> <li>Added consolewrite command to write text into the console window.</li> <li>Added debugwrite for writing text into the debug output.</li> <li>Added ~\$folder.mydocuments\$ variable.</li> </ul>
21/02/2010	2.41	<ul style="list-style-type: none"> <li>Fixed a bug in elevate command: when running a program without parameters, NirCmd sent the elevate command string as a parameter of the program.</li> </ul>
16/02/2010	2.40	<ul style="list-style-type: none"> <li>Added elevate and elevatecmd commands to run and execute commands with administrator rights under Windows 7/Vista/2008.</li> <li>Added monitor parameter to setdisplay command, for using it with multiple monitors.</li> <li>Added setprimarydisplay command.</li> <li>Added folder.programfiles and folder.common_programfiles variables.</li> <li>Added async_off, async_on, async_low for monitor command. You can try them if NirCmd.exe remains in memory when you use monitor on/off command.</li> </ul>



26/05/2009	2.37	<ul style="list-style-type: none"> <li>Improved the help .chm file - Now it contains a link to every command.</li> <li>Added help command - Automatically opens the reference of the desired command in <a href="http://nircmd.nirsoft.net">http://nircmd.nirsoft.net</a>. For example, if you type 'nircmd.exe help speak', the reference page of speak command - <a href="http://nircmd.nirsoft.net/speak.html">http://nircmd.nirsoft.net/speak.html</a> will be opened in your default browser.</li> </ul>
19/05/2009	2.36	<ul style="list-style-type: none"> <li>Fixed bug in script: Sometimes the last line in the script was not executed properly.</li> </ul>
20/04/2009	2.35	<ul style="list-style-type: none"> <li>Added loop command - execute a command multiple times.</li> <li>Added curtime and curdate variables.</li> <li>waitprocess - Added option to execute a NirCmd command after the process was closed.</li> <li>Added rem (or ;) command, just for adding remarks in a script.</li> </ul>
01/04/2009	2.30	<ul style="list-style-type: none"> <li>Added speak command - Speaks the contents of the text or file that you specify, by using the Speech library (SAPI) that comes with Windows XP and Windows Vista.</li> <li>Added savescreenshotfull command.</li> </ul>
02/02/2009	2.25	<ul style="list-style-type: none"> <li>New actions for clipboard command - loadclp and saveclp (load/save in Windows .clp format)</li> <li>New command: setprocessaffinity</li> </ul>
01/01/2009	2.20	<ul style="list-style-type: none"> <li>New commands: waitprocess, setprocesspriority, qboxtop, qboxcomtop.</li> <li>New action in clipboard command: copyimage (Copy image file to the clipboard)</li> </ul>
25/06/2008	2.15	<ul style="list-style-type: none"> <li>New commands: setcursorwin, savescreenshotwin, filldelete</li> </ul>
12/04/2008	2.10	<ul style="list-style-type: none"> <li>New commands: shellcopy, savescreenshot</li> </ul>
09/02/2008	2.05	<ul style="list-style-type: none"> <li>New commands: shellrefresh, convertimage, convertimages</li> <li>New option in clipboard command - saveimage</li> </ul>
16/06/2007	2.00	<ul style="list-style-type: none"> <li>All sound volume commands and monitor command now works under Vista.</li> <li>New commands: dlg, dlgany, sendkey, sendmouse, returnval</li> <li>New actions in win command - dlgclick, dlgsettext, dlgsetfocus</li> </ul>
24/07/2006	1.85	<ul style="list-style-type: none"> <li>New command: cmdshortcutkey</li> </ul>
15/07/2006	1.84	<ul style="list-style-type: none"> <li>Starting from this version, if you don't specify a sound device index in setsysvolume, changesysvolume, setsysvolume2, changesysvolume2 and mutesysvolume commands, the default sound device in control panel ("preferred device") is used.</li> <li>New find options in win command: active, foreground, desktop</li> </ul>
20/05/2006	1.83	<ul style="list-style-type: none"> <li>Fixed bug with parameter variables: parameter dialog-box appear more than once.</li> </ul>

		<ul style="list-style-type: none"> <li>New find option in Win command: alltopnodesktop</li> </ul>
18/02/2006	1.82	<ul style="list-style-type: none"> <li>New win actions: postmsg and sendmsg.</li> <li>New command: closeprocess</li> <li>NirCmd now returns a non-zero value on error.</li> </ul>
19/11/2005	1.81	<ul style="list-style-type: none"> <li>New command: emptybin - Empty the Recycle Bin.</li> <li>New optional parameter in setdisplay command: refresh rate.</li> <li>rashangup command without parameter - disconnect all active connections.</li> </ul>
24/09/2005	1.80	<ul style="list-style-type: none"> <li>New commands: exec2, cmdwait.</li> <li>New options in clipboard command.</li> <li>Volume commands (setsysvolume, changesysvolume, setsysvolume2, changesysvolume2, mutesysvolume) now allows you to change the volume of non-default sound card.</li> </ul>
07/07/2005	1.70	<ul style="list-style-type: none"> <li>New win actions: activate, hideshow, +style, -style, +exstyle, -exstyle</li> <li>New commands: movecursor, setcursor.</li> <li>New command: setfilefoldertime - allows you to change the date/time of folders, under Windows 2000/XP only.</li> </ul>
10/06/2005	1.62	<ul style="list-style-type: none"> <li>New actions in win command: focus, enable, disable</li> </ul>
26/05/2005	1.61	<ul style="list-style-type: none"> <li>new command: sysrefresh - make a general system refresh after modifying your system settings in the Registry.</li> </ul>
07/05/2005	1.60	<ul style="list-style-type: none"> <li>New paramsfile command: Allows you to execute a command multiple times by loading the parameters from a text file.</li> <li>New actions in win command: move, center, settex.</li> <li>New 'child' action in win command: Allows you to move/hide child windows (For example: the start button in the system tray)</li> <li>New clipboard command: Allows you to put a string in the clipboard, and clear the clipboard.</li> <li>New gac command: Allows you to install assemblies in Global Assembly Cache.</li> </ul>
28/12/2004	1.56	<ul style="list-style-type: none"> <li>New variable: ~\$clipboard\$ - paste the text that you copied to the clipboard into the command-line.</li> <li>Parameter variables: ~\$param.ParamName\$ - Display an input window for typing the text that you want insert into the command-line.</li> </ul>
07/12/2004	1.55	Added RegEdit command.
18/11/2004	1.54	<ul style="list-style-type: none"> <li>On reg commands (regsetval, regdelval, regdelkey), you can now also use the following abbreviated key name: HKCU for HKEY_CURRENT_USER, HKLM for HKEY_LOCAL_MACHINE, HKCR for HKEY_CLASSES_ROOT, HKU for HKEY_USERS, HKCC for HKEY_CURRENT_CONFIG.</li> </ul>

		<ul style="list-style-type: none"> <li>redraw action for win command.</li> </ul>
03/11/2004	1.53	Fixed 'monitor off' problem in XP SP2.
09/10/2004	1.52	rasapi32.dll is now loaded only when using the dial commands (rasdial, rasdialdlg, rashangup). In previous versions, NirCmd was statically linked to rasapi32.dll, and that caused problems in old NT systems.
29/09/2004	1.51	Variable names are now enclosed with '\$' char instead of '%' char. I made this change because the '%' char causes problems when running NirCmd from cmd/bat file. using the '%' char for variable names is still supported for backward compatibility.
22/09/2004	1.50	<ul style="list-style-type: none"> <li>New commands: cmdshortcut, regsvr, mutesysvolume, changesysvolume, changesysvolume2, setsysvolume2.</li> <li>New actions in win command: trans, setsize, settopmost, itile.</li> <li>New variables: folder.nircmd, nir.exe file</li> <li>nircmdc.exe - console version of NirCmd.</li> <li>Changed name - from NirComLine to NirCmd.</li> </ul>
05/05/2004	1.40	<ul style="list-style-type: none"> <li>New commands: wait, setvolume, setsysvolume, execcmd, exec.</li> <li>New commands for executing NirCmd on remote machines: remote and multiremote.</li> <li>Folder and system variables.</li> <li>New option in setfiletime command - set the file time to the current time (now parameter)</li> <li>New parameter in shortcut command - hot key.</li> </ul>
01/03/2004	1.30	<ul style="list-style-type: none"> <li>New commands: qboxcom, beep, stdbeep, cdrom, urlshortcut, monitor, screensaver, standby, and hibernate.</li> <li>Fixed the bug with ~x string sequences.</li> </ul>
08/01/2004	1.20	<ul style="list-style-type: none"> <li>New commands: setdialuplogon, script</li> <li>Special string sequences (~q, ~t, ~n and so on) to allow you to embed quotes and new line characters into the parameters of any command.</li> </ul>
19/11/2003	1.11	New options in shortcut command
10/10/2003	1.10	New commands: killprocess, service, memdump, win, lockws.
09/09/2003	1.00	First release.

## License

This utility is released as freeware. You are allowed to freely distribute this utility via floppy disk, CD-ROM, Internet, or in any other way, as long as you don't charge anything for this. If you distribute this utility, you must include all files in the distribution package, without any modification !

## Using NirCmd utility

This utility is a standalone executable, and it doesn't require any installation process or additional DLLs. just copy the executable to any folder you want, and run it with the desired command-line option, according to your needs. It's recommended to copy the executable of NirCmd (nircmd.exe) to your windows directory, or to any other folder listed in your PATH environment variable, so you won't need to type the full path of nircmd in each time that you want to use it.

Starting from version 1.50, console version of NirCmd (nircmdc.exe) is also available. In the console version, all error messages goes to the console window, instead of using a message box.

## General Syntax Of NirCmd

**nircmd.exe {showerror} [command] [command parameters]**

The **showerror** parameter is optional. If you specify this parameter, an error message will be displayed on the screen if an error is occurred during the execution of the command. Otherwise, any error will be ignored. For example:

```
nircmd.exe showerror rasdial "diall "
```

```
nircmd.exe rasdial "diall "
```

In above 2 examples, NirCmd will try to dial with the "diall" entry. If the dialing process is failed on the first example, an error message describing the problem will be displayed. If the dialing process is failed on the second example, error message won't be displayed.

## Special String Sequences

Starting from version 1.20, NirCmd allows you to insert special characters into any parameter in NirCmd commands. The following table describes the special string Sequences that you can use:

Sequence	Description
~n	New line characters (CR-LF)
~q	Double-quote character: "
~t	Tab character
~xnn	Allows you to represent any character by providing its ascii code in 2-digit Hexadecimal number. For Example: ~x41 ('A' character), ~x27 (single-quote character)
~\$variable\$	Represents a system variable. For more information about this option, <a href="#">read below</a>
~~	Represents a single '~' character.

Example:

```
infobox "This is the first line~n~qThis is a second line, in quotes~q" "Example"
```

The above example displays a message-box containing 2 lines.

## System Variables

Starting from version 1.40, NirCmd provides special variables that represents folders and other variables on your system. Each variable starts with '~' character, followed by the variable name enclosed with '\$' characters. For example: ~\$folder.desktop\$ represents your desktop folder.

The following table specifies the variables that you can use:

Variable Name	Description
clipboard	Represents the text that you copied into the clipboard.
param.paramname	When you specify a param variable, an input window will appear on the screen, and will ask you to type the text for this variable. The text that you type will be inserted into the command line. For example: in the following command, you will be asked to type 2 parameters (1 and 2), and the text that you type will appear in the message box. nircmd infobox ~\$param.1\$ ~\$param.2\$
fparam.number	Specifies a parameter loaded from a text file. For more information, see <a href="#">paramsfile command</a>
sys.varname	You can use this option to represent any environment variable on your system. For example: ~\$sys.username\$, ~\$sys.windir\$, ~\$sys.computername\$, ~\$sys.logonserver\$, ~\$sys.temp\$, and so on...
nir.exe file	Represents the full-path exe file of the current running NirCmd (For example: c:\winnt\nircmd.exe)
loopcount	Specifies the loop count number when using the loop command.
curtime.TimeFormat	Specifies the current time, according to the format specified in 'TimeFormat' section. For example: ~\$curtime.HH_mm_ss\$ For more information about the format string, see curtime format section.
curdate.DateFormat	Specifies the current date, according to the format specified in 'DateFormat' section. For example: ~\$curdate.MM_dd_yyyy\$ For more information about the format string, see curdate format section.
folder.nircmd	Represents the folder the NirCmd is running from.
folder.desktop	Represents the user's desktop folder.
folder.start_menu	Represents the user's start-menu folder.
folder.programs	Represents the user's programs folder (under the start-menu folder).
folder.startup	Represents the user's startup folder.
folder.recent	Represents the user's recent folder.
folder.favorites	Represents the user's favorites folder.
folder.cookies	Represents the user's cookies folder.
folder.appdata	Represents the user's Application Data folder. (For example: C:\Documents and Settings\Administrator\Application Data)
folder.common_desktop	Represents the common desktop folder.

folder.common_start_menu	Represents the common start menu folder.
folder.common_programs	Represents the common programs folder.
folder.common_startup	Represents the common startup folder.
folder.common_favorites	Represents the common favorites folder.
folder.windows	Represents the windows folder. (e.g.: C:\Windows)
folder.system	Represents the system folder. (e.g.: C:\Windows\System32)
folder.programfiles	Represents the Program Files folder. (e.g.: C:\Program Files)
folder.common_programfiles	Represents the common Program Files folder. (e.g.: C:\Program Files\Common)
folder.mydocuments	Represents the 'My Documents' folder.

Examples:

- Create a shortcut under the start menu-programs folder:  
**shortcut "c:\winnt\system32\calc.exe" "~\$folder.programs\$\My Programs" "Calculator"**
- Delete a shortcut in the user's desktop:  
**execmd del "~\$folder.desktop\$\calc.lnk"**
- Display the current logged-on user:  
**infobox "The current logged-on user is ~\$sys.username\$" "logged-on user"**
- Save a screenshot in filename containing the current time and date  
**savescreenshot c:\temp\scr~\$currdate.MMddyyyy\$~\$currtime.HHmms\$.png**

## currtime Format String

The time format string may contain the following elements: (The format string is case sensitive !)

h	Hours with no leading zero for single-digit hours; 12-hour clock.
hh	Hours with leading zero for single-digit hours; 12-hour clock.
H	Hours with no leading zero for single-digit hours; 24-hour clock.
HH	Hours with leading zero for single-digit hours; 24-hour clock.
m	Minutes with no leading zero for single-digit minutes.
mm	Minutes with leading zero for single-digit minutes.
s	Seconds with no leading zero for single-digit seconds.
ss	Seconds with leading zero for single-digit seconds.
t	One character time-marker string, such as A or P.
tt	Multicharacter time-marker string, such as AM or PM.

## currdate Format String

The date format string may contain the following elements: (The format string is case sensitive !)

d	Day of month as digits with no leading zero for single-digit days.
dd	Day of month as digits with leading zero for single-digit days.
ddd	Day of week as a three-letter abbreviation.
dddd	Day of week as its full name.
M	Month as digits with no leading zero for single-digit months.
MM	Month as digits with leading zero for single-digit months.
MMM	Month as a three-letter abbreviation.
MMMM	Month as its full name.
y	Year as last two digits, but with no leading zero for years less than 10.
yy	Year as last two digits, but with leading zero for years less than 10.
yyyy	Year represented by full four digits.

## Executing NirCmd commands on remote computers

Starting from version 1.40, you can execute all NirCmd commands in remote computers. However, you should be aware of the following restrictions and problems:

- This option is only available for Windows NT, Windows 2000 and Windows XP. You cannot use it on Windows 9x machines.
- You must log on to the remote machine as administrator.
- The Schedule service in the remote machine must be started. (By default, the Schedule service is automatically started in all NT/2000/XP machines, so you don't have to worry too much about this)
- Due to some limitations of the Schedule service, you have to wait up to 60 seconds until the command is actually executed in the remote machine.
- The command in the remote machine is executed under 'SYSTEM' account, and thus some commands may not work in the same way as you execute them locally.

There are 2 commands for executing NirCmd in remote machines:

- [remote](#) - for executing command on a single remote computer.
- [multiremote](#) - for executing command on multiple remote computers.

## Help Command

If you don't remember the exact syntax of a command, you can always look in the help file of NirCmd (NirCmd.chm) or you can use the help command to get the command reference in <http://nircmd.nirsoft.net>. For example:  
nircmd.exe help multiremote

## NirCmd Commands

```
remote {copy} [\computer] [NirCmd command]
```

Executes NirCmd command in a remote computer. if **{copy}** parameter is specified, NirCmd will copy itself to the Windows directory of the remote machine. You must specify this parameter if NirCmd doesn't exist in the Windows directory of the remote machine.

Examples:

```
remote copy \\comp1 exitwin poweroff
remote \\comp2 infobox "This is a message from NirCmd" "Hello !"
remote \\comp3 lockws
```

#### **multiremote {copy} [computers file] [NirCmd command]**

Executes NirCmd command in multiple remote machines. if **{copy}** parameter is specified, NirCmd will copy itself to the Windows directory of the remote machines. You must specify this parameter if NirCmd doesn't exist in the Windows directory of the remote machines.

The **[computers file]** parameter is a file that contains the computers list separated by Enter characters (CRLF).

Example for computers list file:

```
\\comp1
\\comp2
\\winxp1
\\winxp2
\\nirsoft
```

The command specified in **[NirCmd command]** will be executed in all computers specified in the computers file.

Examples:

- Turn off all computers specified in computers.txt  
**multiremote "c:\temp\computers.txt" exitwin poweroff force**
- Lock all computers specified in computers.txt  
**multiremote "c:\temp\computers.txt" lockws**
- Copy NirCmd to all computers specified in computers.txt, and then turn off their monitors.  
**multiremote copy "c:\temp\computers.txt" monitor off**
- Copy NirCmd to all computers specified in computers.txt, and run WinUpdatesList utility in order to save Windows Updates info in all computers.  
**multiremote copy "c:\temp\computers.txt" exec hide c:\temp\wul.exe /shtml  
"c:\temp\wul\_~\$sys.computername\$.html"**

#### **cmdwait [Number of Milliseconds] [NirCmd Command]**

Wait the specified number of Milliseconds, and then execute the specified NirCmd command.

Example:

```
cmdwait 1000 monitor async_off
```

#### **loop [Number of Times] [Number of Milliseconds] [NirCmd Command]**

Execute a NirCmd mutple times, in a loop. Between the command executions, NirCmd will wait the number of Milliseconds specified in the second parameter. You can use the ~\$loopcount\$ variable to specify the call number in the loop (from 1 to n).



Examples:

```
loop 3 2000 speak text "Call Number ~$loopcount$"
loop 10 60000 savescreenshot c:\temp\scr~$loopcount$.png
```

#### **paramsfile [Parameters File] [Delimiters] [Quote Character] [NirCmd Command]**

This powerfull command allows you to execute NirCmd Command multiple times, by loading one or more parameters from a text file. The text file can be comma-delimited, tab-delimited, semicolon-delimited, or delimited by any char that you specify in [Delimiters] parameter.

Here's an example... Let's say that you have the following text file, containing user names and passwords, delimited by comma:

```
user01, 123456
user02, abcfg
user03, 5fr23
user04, 33333
```

The following NirCmd commands sequence will create 4 users with the specified user-names and passwords (by using 'net user' command provided by Windows 2000/XP operating system)

```
nircmd.exe paramsfile "c:\temp\users.txt" "," "" execmd net user ~$fparam.1$ ~$fparam.2$ /add
```

The ~\$fparam.1\$ specifies the first entry in each line - the user name. The ~\$fparam.2\$ specifies the second entry in each line - the password.

Examples:

```
paramsfile "c:\temp\folders.txt" "" "" execmd md ~$fparam.1$
paramsfile "c:\temp\shortcuts.txt" "," "~q" shortcut ~$fparam.1$ "~$folder.desktop$" ~$fparam.2$
```

#### **infobox [message text] [title]**

Displays a simple message box on the screen.

Example:

```
infobox "This is a message box !" "message"
```

#### **qbox [message text] [title] [program to run]**

Displays a question dialog-box on the screen. If the user answers "Yes", run a program

Example:

```
qbox "Do you want to run the calculator ?" "question" "calc.exe"
```

#### **qboxtop [message text] [title] [program to run]**

Similar to qbox, but displays the message-box as top-most window.

#### **qboxcom [message text] [title] [NirCmd Command]**

Displays a question dialog-box on the screen. If the user answers "Yes", execute the NirCmd command in the third parameter.

Examples:

```
qboxcom "Do you want to lock your workstation ?" "question" lockws
qboxcom "Do you want to turn off the monitor ?" "monitor" monitor off
```

**qboxcomtop [message text] [title] [NirCmd Command]**

Similar to qboxcom, but displays the message-box as top-most window.

**script [Script File]**

This command allows you to execute a sequence of commands stored inside the file specified in **[Script File]** parameter.

Example:

**script "c:\temp\msg.ncl"**

Example for a script:

**infobox "Hello !" "This is the first message"**

**infobox "Hello !" "This is the second message"**

**infobox "Hello !" "This is the third message"**

**regedit [key] [value name]**

Opens RegEdit with the specified key/value.

Examples:

**regedit "HKLM\Software\Microsoft\Windows\CurrentVersion" "CommonFilesDir"**

**regedit "HKEY\_CURRENT\_USER\Control Panel\Desktop" "Wallpaper"**

**regsetval [type] [key] [value name] [value]**

Writes a value to the Registry. the **[type]** parameter can be "sz" or "expand\_sz" for strings, "dword" for 32-bit number, or "binary" for binary data. If the key you specify in **[key]** parameter doesn't exist, it'll automatically be created.

Examples:

**regsetval dword "HKEY\_CURRENT\_USER\Software\Test" "Value1" "0xaabbccdd"**

**regsetval dword "HKCU\Software\Test" "Value2" "10561"**

**regsetval sz "HKCU\Software\Test" "Value3" "string value"**

**regsetval binary "HKLM\Software\Test" "Value4" "6a cb ab dd 80 86 95"**

**regdelval [key] [value]**

Deletes a value from the Registry.

Example:

**regdelval "HKEY\_CURRENT\_USER\Software\Test" "Value1"**

**regdelkey [key]**

Deletes entire key from the Registry. This command will failed if the specified key has one or more subkeys.

Example:

**regdelkey "HKEY\_CURRENT\_USER\Software\Test"**

**inisetval [INI filename] [section] [key] [value]**

Writes a value into INI file. If the file doesn't exist, it'll automatically be created.

Example:

**inisetval "c:\windows\test.ini" "section1" "TestValue" "1"**

**inidelval [INI filename] [section] [key]**

Deletes a key-value pair inside INI file.

Example:

```
inidelval "c:\windows\test.ini" "section1" "TestValue"
```

**inidelsec [INI filename] [section]**

Deletes entire section in INI file, including all values in it.

Example:

```
inidelsec "c:\windows\test.ini" "section1"
```

**regsvr [reg or unreg] [filename or wildcard] [log file]**

Register or unregister DLL/OCX files.

In the first parameter, specify "reg" for registering files, or "unreg" for unregistering files.

In the second parameter, you can specify a single file that you want to register or unregister, or multiple files, by using wildcard.

The third parameter is optional, and specifies a log filename for saving the results of the registering operation.

Examples:

```
regsvr reg "f:\winnt\system32\fv*.ocx" "c:\temp\log1.txt"
```

```
regsvr unreg "f:\winnt\system32\ab*.dll"
```

**rasdial [Entry Name] {Phonebook File} {User Name} {Password} {Domain}**

Dials to RAS entry (VPN or phone line), without displaying any user interface. Except the [Entry Name] parameter, all other parameters are optional. If you don't specify the optional parameters or you specify empty string (""), the default values are used.

Examples:

```
rasdial "My Internet"
```

```
rasdial "VPN1" "" "user1" "abc1234" "domain1"
```

**inetdial [Entry Name]**

Dials to RAS entry (VPN or phone line), and display informational dialog-box during the dialing process.

This command is only supported If Internet Explorer version 4.0 or greater is installed. Be aware that this command might not work on old versions of Windows 98.

Examples:

```
inetdial "VPN1"
```

**rasdialdlg [Entry Name] {Phonebook File} {Phone Number or IP Address}**

Displays a RAS entry for dialing. The {Phonebook File} and {Phone Number} parameters are optional. If you don't specify the {Phonebook File} parameter or you specify an empty string, the default phonebook is used.

This command is only supported on Windows NT, Windows 2000, and Windows XP. It's not supported on Windows 98/ME.

Examples:

```
rasdialdlg "Internet1" "" "9876543"
```

```
rasdialdlg "VPN1"
```

**rashangup {Entry Name}**

Hangs up a RAS connection. You can use this command to disconnect a connection that you previously created with `instdial` or `rasdial` commands. If you don't specify the `{Entry Name}` parameter, all active connections will be disconnected.

Examples:

```
rashangup "VPN1"  
rashangup
```

### `setdialuplogon [Entry Name] [User Name] {Password} {Domain} {Phonebook File}`

Modifies the user name, password and domain of a dial-up item.

The parameters:

- **[Entry Name]**: The name of the dial-up entry
- **[User Name]**: The user-name to set into the specified dial-up entry.
- **{Password}**: The password to set into the specified dial-up entry.
- **{Domain}**: The domain name to set into the specified dial-up entry.
- **{Phonebook File}**: The phonebook file containing the entry you want to change. If you don't specify this parameter, the default phonebook is used.

Examples:

```
setdialuplogon "VPN1" "user1" "asder443" "domain1"  
setdialuplogon "Internet1" "nirs3" "q5r4df7yh5"
```

### `exitwin [shutdown type] {additional option}`

This command terminates the current session of Windows. The **[shutdown type]** parameter can be one of the following values:

- **logoff**: Shut down all running processes, log off the current user, and display the log on dialog to allow another user to log into the system.
- **reboot**: Shut down the entire system, and then reboot.
- **poweroff**: Shut down the entire system, and then turn off the power. (Only for systems that support this feature !)
- **shutdown**: Simply shut down the entire system, without reboot and without turning the power off.

You can also use one of the following additional options:

- **force**: Forces all applications to terminate immediately. Using this option can caused the current running application to lose data. Use it only in extreme cases !
- **forceifhung**: Forces applications to terminate if they are not responding. This option is only available in Windows 2000/XP.

Examples:

```
exitwin reboot  
exitwin poweroff force  
exitwin logoff forceifhung
```

### `initshutdown {message} {timeout value in seconds} {force} {reboot}`

Initiate a system shutdown. This command is somewhat similar to `exitwin` command, but it has some other features that `exitwin` doesn't support. This command is not supported on Windows 95/98/ME. All parameters of this

command are optional. If you don't specify any parameter, the system shutdown is initiated without displaying message and without waiting.

The parameters:

- **{message}**: The message to display before starting the system shut down.
- **{timeout value in seconds}**: The number of seconds to wait before starting the system shut down.
- **{force}**: Forces all applications to terminate, even without saving your work.
- **{reboot}**: Reboot after the system shutdown.

Examples:

**initshutdown "shutting down the system within 60 seconds" 60 reboot**

**initshutdown "" 0 force**

### **abortshutdown**

Abort the system shutdown that initiated with initshutdown command. This command doesn't accept any additional parameters.

Example:

**abortshutdown**

### **lockws**

Lock the workstation. After locking your workstation, you must type your password in order to continue your work. This command is only available on Windows 2000 and Windows XP.

### **monitor [action]**

Changes the state of the display monitor. The **[action]** parameter may contain the following values:

- **off**: Turn off the monitor
- **async\_off**: Turn off the monitor. Use this value if NirCmd remains in memory when using monitor off.
- **on**: Turn on the monitor
- **low**: Set the monitor to low power state.

This command only works in systems that support this feature. If you have a problem that NirCmd remains in memory when using this command, you may try to use **async\_off**, **async\_on** and **async\_low** actions instead of **on/off/low** actions.

Examples:

**monitor off**

**monitor on**

**monitor async\_off**

### **screensaver**

Starts the default screen saver.

### **screensavertimeout [Timeout]**

Specifies the number of seconds that the system will wait before starting the screensaver.

Examples:

**screensavertimeout 30**  
**screensavertimeout 90**

#### **standby {force}**

Shuts down the computer and puts it in 'standby' mode. If the **{force}** parameter is specified, the system is forced to go into standby mode without asking for permission from all applications. Generally, it's not recommended to use this parameter.

This command works only in systems that supports standby mode.

Example:

**standby force**  
**standby**

#### **hibernate {force}**

Shuts down the computer and puts it in 'hibernate' mode. If the **{force}** parameter is specified, the system is forced to go into hibernate mode without asking for permission from all applications. Generally, it's not recommended to use this parameter.

This command works only in systems that supports hibernate mode.

Example:

**hibernate force**  
**hibernate**

#### **shortcut [filename] [folder] [shortcut title] {arguments} {icon file} {icon resource number} {ShowCmd} {Start In Folder} {Hot Key}**

Creates a shortcut to a file.

The parameters:

- **[filename]**: Create a shortcut to this filename.
- **[folder]**: Specify the destination folder that inside it the shortcut will be created. You can specify any valid folder, including the special variables that represent system folders, like ~\$folder.desktop\$ (Desktop folder), ~\$folder.programs\$ (Start-Menu-Programs folder), and so on...
- **[shortcut title]**: The text displayed in the shortcut.
- **{arguments}**: Optional parameter - Additional arguments to execute the filename.
- **{icon file}**: Optional parameter - Use this parameter if your want that the shortcut will be displayed with icon other than the default one.
- **{icon resource number}**: Optional parameter - The resource number inside the icon file.
- **{ShowCmd}**: Optional parameter - Use this parameter if you want to maximize or minimize the window of the program. Specify "max" to maximize the window or "min" to minimize it.
- **{Start In Folder}**: Optional parameter - Specifies the "Start In" folder. If you don't specify this parameter, the "Start In" folder is automatically filled with the folder of the program you specify in **[filename]** parameter.
- **{Hot Key}**: Optional parameter - Specifies an hot-key that will activate the shortcut. For example:  
Alt+Ctrl+A, Alt+Shift+F8, Alt+Ctrl+Shift+Y

Examples:

**shortcut "f:\winnt\system32\calc.exe" "~\$folder.desktop\$" "Windows Calculator"**

```

shortcut "f:\winnt\system32\calc.exe" "~$folder.programs$\Calculators" "Windows Calculator"
shortcut "f:\Program Files\KaZaA\Kazaa.exe" "c:\temp\MyShortcuts" "Kazaa"
shortcut "f:\Program Files" "c:\temp\MyShortcuts" "Program Files Folder" "" "f:\winnt\system32\shell32.dll" 45
shortcut "f:\Program Files" "c:\temp\MyShortcuts" "Program Files Folder" "" "" "" "max"

```

#### **cmdshortcut [folder] [shortcut title] [command]**

Creates a shortcut to NirCmd command. When you activate the shortcut (by doubleclicking it), The command that you specify in **[command]** parameter will be executed.

Examples:

```

cmdshortcut "~$folder.desktop$" "Open CDROM" cdrom open k:
cmdshortcut "c:\temp" "Turn Monitor Off" monitor off

```

#### **cmdshortcutkey [folder] [shortcut title] [hot key] [command]**

Similar to cmdshortcut command, but also allows you to specify an hot key to activate the command.

Examples:

```

cmdshortcutkey "~$folder.desktop$" "Open CDROM" "Ctrl+Shift+K" cdrom open k:
cmdshortcutkey "c:\temp" "Turn Monitor Off" "Ctrl+Shift+M" monitor off

```

#### **urlshortcut [URL] [Folder] [Shortcut Title]**

Creates a shortcut to URL address. (Like the shortcuts in Favorites of Internet Explorer) The **[Folder]** parameter specifies the destination folder that inside it the shortcut will be created. If you want to create the shortcut inside your default Favorites folder, use ~\$folder.favorites\$ variable.

Examples:

```

urlshortcut "http://www.nirsoft.net" "~$folder.favorites$" "NirSoft Web Site"
urlshortcut "http://www.google.com" "~$folder.favorites$\Search Engines" "Google"
urlshortcut "http://www.yahoo.com" "c:\temp" "Yahoo"

```

#### **exec [show/hide/min/max] [application + command-line]**

Runs an application, and optionally specify one or more command-line parameters for the executed application. The **[show/hide/min/max]** parameter specifies whether the running application will be visible or not. If 'hide' is specified, the running application won't be visible to the user. If 'max' is specified, the running application window will be maximized. If 'min' is specified, the running application window will be minimized.

Example:

```

exec show "f:\winnt\system32\calc.exe"
exec hide "c:\temp\wul.exe" /savelangfile
exec max "c:\temp\test.exe"

```

#### **exec2 [show/hide/min/max] [working folder] [application + command-line]**

Similar to exec command, but also provide another parameter, [working folder], that specifies the default working folder for the application that you run.

Example:

```

exec2 show "f:\winnt\system32" "f:\winnt\system32\calc.exe"
exec2 hide c:\temp "c:\temp\wul.exe" /savelangfile

```

#### **execmd [command]**

Executes a command of Command-Prompt (Console Window) without displaying anything on the screen.

Examples:

```
execmd del "%folder.desktop%\calc.lnk"
execmd mkdir f:\temp\desktop1
execmd copy "%folder.desktop%\*.lnk" f:\temp\desktop
```

#### shexec [operation] [filename]

Opens or Prints a filename according to the definition of the file extension in the Registry. For example: a filename with .doc extension will be opened with MS-Word, a filename with .html extension will be opened with your default browser. If you specify a valid web site address, the Web site will be opened in your default browser. the [operation] parameter can be "open" or "print".

Examples:

```
shexec "open" "c:\my documents\abc.doc"
shexec "print" "d:\files\1.pdf"
shexec "open" "http://www.nirsoft.net"
```

#### setfiletime [filename or wildcard] [Created Date] {Modified Date} {Accessed Date}

Changes the Created/Modified/Accessed date of one or more files. The first parameter can be a single filename or wildcard string. The dates parameters must be specified in the following format: "dd-mm-yyyy hh:mm:ss" if a date parameter is not specified or you specify an empty string (""), the date won't be changed. If you specify "now" as date parameter, the current date and time will be used

Examples:

```
setfiletime "c:\temp\*.exe" "15-05-2004 11:22:12" "22-01-2005 21:22:55"
setfiletime "c:\temp\myfile.txt" "24-06-2003 17:57:11" "" "22-11-2005 10:21:56"
setfiletime "c:\temp\myfile.txt" now now
```

#### setfilefoldertime [filename, folder or wildcard] [Created Date] {Modified Date} {Accessed Date}

Changes the Created/Modified/Accessed date of one or more files/folders. This command is very similar to setfiletime command. The only difference is that setfilefoldertime command can also change the date of a folder - only under Windows 2000/XP.

Examples:

```
setfilefoldertime "c:\temp\myfolder" "10-01-2003 11:22:12" "12-11-2002 02:34:11"
setfilefoldertime "c:\temp\myfolder" now now
```

#### clonefiletime [source filename] [Wildcard]

Clone the Created/Modified/Accessed time of one file into one or more files. The Created/Modified/Accessed time is taken from the file specified in [source filename] parameter and set into the file or wildcard specified in [Wildcard] parameter.

Examples:

```
clonefiletime "c:\temp\file1.txt" "c:\files\*.txt"
clonefiletime "c:\temp\file1.txt" "c:\temp\file2.txt"
```

#### moverecyclebin [Wildcard]

Moves one or more files to the Recycle Bin of Windows. You can specify a single file, or multiple files using wildcard.

### Warning !!

When using wildcards, this command also moves subfolders into the Recycle Bin, so use it very carefully !



Example:  
**move recyclebin f:\temp\myfile.txt**  
**move recyclebin f:\temp\1\\*.txt**

#### **emptybin {drive}**

Empty the Recycle Bin. the {drive} parameter specifies the drive of the Recycle Bin that you want to empty. If the {drive} parameter is not specified, all Recycle Bin on all drives will be emptied.

Example:  
**emptybin f:**  
**emptybin**

#### **setdisplay {monitor:index/name} [width] [height] [color bits] {refresh rate} {-update reg} {-allusers}**

Changes your display settings. The [width] and [height] parameters represents the number of pixels on your screen. The [color bits] parameter represents the number of colors shown on your screen (8 - 256 color, 16 - 16bit color, 24 - 24bit color, and so on). {refresh rate} is an optional parameter that specifies the monitor refresh rate. If you specify the {-update reg} parameter, the new settings will be saved in the Registry. If you specify both {-update reg} and {-allusers} parameters, the new settings will be saved in the Registry for all users.

If you have multiple monitors, you can use the optional monitor parameter, which specifies for which monitor you want to change the display settings. You can specify the monitor by index (0 for the first monitor, 1 for the second one, and so on) or by specifying a string in the system monitor name. The monitor name can be found in the Device manager of Windows: Right click on the monitor item, and then choose 'Properties'. The string displayed in the 'location' field is the monitor name.

Example:  
**setdisplay 800 600 24 -update reg**  
**setdisplay 1024 768 24 90**  
**setdisplay 1024 768 8**  
**setdisplay monitor:1 1024 768 24 90**  
**setdisplay monitor:name1 1024 768 24 90**

#### **setprimarydisplay [Monitor index/name]**

Set the primary monitor, for systems with multiple monitors only.

Example:  
**setprimarydisplay 1**  
**setprimarydisplay 2**  
**setprimarydisplay \\.\DISPLAY3**

#### **setbrightness [level] {mode}**

This command only works on laptop/netbook computers. It allows you to set the brightness of the laptop screen. The [level] parameter is brightness value between 0 and 100. 0 = very dark, 100 = very bright.

{mode} is optional parameter that may contain the following values:

- 1 = Change the brightness under AC power.
- 2 = Change the brightness under DC power.
- 3 = Change the brightness for both AC power and DC power (The default).

Examples:  
**setbrightness 50**  
**setbrightness 40 1**  
**setbrightness 60 2**

#### **changebrightness [level] {mode}**

This command only works on laptop/netbook computers. It allows you to change the brightness of the laptop screen. The **[level]** parameter is brightness value to change. positive value increases the brightness, and negative value decreases the brightness.

**{mode}** is optional parameter that may contain the following values:

1 = Change the brightness under AC power.

2 = Change the brightness under DC power.

3 = Change the brightness for both AC power and DC power (The default).

Examples:

**changebrightness 7**

**changebrightness -10**

**changebrightness 5 2**

### **killprocess [process]**

Terminates the specified process. In the **[process]** parameter, you can specify the full path of the process filename (e.g. 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

Examples:

**killprocess iexplore.exe**

**killprocess "c:\winnt\system32\calc.exe"**

**killprocess /1830**

### **closeprocess [process]**

Terminates the specified process, by closing its top level windows. As opposed to killprocess command, closeprocess doesn't force the process to be closed immediately. Instead, it send WM\_CLOSE message to all top level windows of the specified process. However, be aware that this command will not work on process with no user interface.

In the **[process]** parameter, you can specify the full path of the process filename (e.g. 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

Examples:

**closeprocess iexplore.exe**

**closeprocess "c:\winnt\system32\calc.exe"**

**closeprocess /1830**

### **restartexplorer**

Restarts Windows Explorer gracefully.

### **waitprocess [process] {Command}**

Wait until the specified process is closed.

In the **[process]** parameter, you can specify the full path of the process filename (e.g. 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

Optionally, you can also specify a command to execute after the process was closed.

Examples:

**waitprocess iexplore.exe**

```
waitprocess firefox.exe speak text "Firefox was closed"
waitprocess "c:\winnt\system32\calc.exe"
waitprocess /1830
```

### **suspendprocess [process]**

Suspends the specified process.

In the **[process]** parameter, you can specify the full path of the process filename (e.g: 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

Examples:

```
suspendprocess myprocess.exe
suspendprocess "c:\winnt\system32\calc.exe"
suspendprocess /1130
```

### **resumeprocess [process]**

Resumes the specified process.

In the **[process]** parameter, you can specify the full path of the process filename (e.g: 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

Examples:

```
resumeprocess myprocess.exe
resumeprocess "c:\winnt\system32\calc.exe"
resumeprocess /1250
```

### **setprocesspriority [process] [priority]**

Set a new priority for the specified process.

In the **[priority]** parameter, you can specify one of the following values: normal, low, belownormal, abovenormal, high, realtime.

In the **[process]** parameter, you can specify the full path of the process filename (e.g: 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

Examples:

```
setprocesspriority iexplore.exe high
setprocesspriority "c:\winnt\system32\calc.exe" low
setprocesspriority /1830 abovenormal
```

### **setprocessaffinity [process] [list of processor numbers]**

Set the process affinity, which means - specifying the processors that the process is allowed to use (Only relevant for computers with more than one processor)

In the **[process]** parameter, you can specify the full path of the process filename (e.g: 'C:\Program Files\Internet Explorer\iexplore.exe') or the name of the process only, without the path (e.g.: iexplore.exe). You can also specify the Process ID, by adding the '/' character as a prefix (e.g. /1120).

In the **[list of processor numbers]** parameter, you can specify the processor numbers that you want to use. 0 is the first processor, 1 is the second one, and so on.

Examples:

```
setprocessaffinity iexplore.exe 0 1
setprocessaffinity "c:\winnt\system32\calc.exe" 0 1 2 3
setprocessaffinity /1830 0
```

**memdump [process] [dump file] [bytes per line] {bytes to read} {start address} {nohex} {noascii}**

Reads the memory of process according to the parameters you specify, and writes it into a text file.

The parameters:

- **[process]**: The process that you want to dump its memory. You can specify the full path of the process filename, the process name only, or the Process ID - by adding the '/' character as prefix (e.g.: /520)
- **[dump file]**: The filename to save the memory content of the specified process.
- **[bytes per line]**: Number of memory bytes per line in the saved dump file.
- **{bytes to read}**: Number of bytes to read.
- **{start address}**: Start the dumping process from this memory address.
- **{nohex}**: Optional. If you specify this parameter, the memory data won't be saved in Hexadecimal format.
- **{noascii}**: Optional. If you specify this parameter, the memory data won't be saved in Ascii format.

Examples:

```
memdump iexplore.exe "c:\temp\dump.txt" "16" "0x10000" "0x400000"
memdump calc.exe "c:\temp\calc.txt" "32" "0x30000" "0x400000" nohex
memdump /525 "c:\temp\dump.txt" "16" "0x100000" "0x120000" noascii
```

**service {\Computer} [action] [service name]**

Changes the status and the startup type of services and drivers.

The parameters:

- **{\Computer}**: Optional. Specifies the name of the remote computer. If you omit this parameter, the local computer is used.
- **[action]**: The action you want to do. This parameter may contain one of the following values:
  - **start**: Starts the specified service or driver.
  - **stop**: Stops the specified service or driver.
  - **pause**: Pauses the specified service or driver.
  - **continue**: Continues the specified service or driver.
  - **restart**: Stops and then starts again the specified service or driver.
  - **auto**: Changes the startup type of the specified service or driver to Automatic.
  - **manual**: Changes the startup type of the specified service or driver to Manual.
  - **disabled**: Changes the startup type of the specified service or driver to Disabled.
  - **boot**: (Only for drivers) Changes the startup type of the specified driver to 'Boot'.
  - **system**: (Only for drivers) Changes the startup type of the specified driver to 'System'.
- **[service name]**: The name of the service or driver.

Examples:

```
service start schedule
service \\remote stop schedule
service restart w3svc
```

**win [action] [find] [window to find] [Additional Parameters]**

For child windows:

**win child [find] [top window to find] [action] [find] [child window to find] [Additional Parameters]**

This command allows you to close, hide, show, maximize, and minimize the specified window.

The parameters:

- **[action]:** The action you want to do on the specified window:
  - **child:** Allows you to make an action on child window of the specified top-level window.
  - **close:** Closes the specified windows.
  - **hide:** Hides the specified windows.
  - **show:** Shows again the specified windows. (after hiding it with 'hide' command)
  - **hideshow:** Hides the specified windows and then shows it again. You can use this action to repaint the content of a window.
  - **togglehide:** Toggles the specified windows between visible and hidden state.
  - **activate:** Bring the specified window to the front.
  - **flash:** Flashes the specified window. the values in **[Additional Parameters]** specifies the number of flashes (the default is 5) and the number of milliseconds of every flash.
  - **max:** Maximizes the specified windows.
  - **min:** Minimizes the specified windows.
  - **normal:** Restores the specified windows to normal state, after minimizing or maximizing them.
  - **togglemin:** Toggles the specified windows between minimized and normal state.
  - **togglemax:** Toggles the specified windows between maximized and normal state.
  - **trans:** Make the specified windows transparent. the value in **[Additional Parameters]** is a number between 0 and 255 that specifies the transparency level. 0 = completely transparent. 255 = completely opaque.
  - **setsize:** Set the size of the specified windows. the values in **[Additional Parameters]** specifies the desired window size - x, y, width, height.
  - **move:** move/resize the window. the values in **[Additional Parameters]** specifies the number of pixels to move/change: x, y, width, height.
  - **center:** Center the specified windows.
  - **settopmost:** Set the top-most state of the specified windows. If the value in **[Additional Parameters]** is 1, the specified windows will become top-most windows, if the value is 0, the top-most state will be canceled.
  - **redraw:** Redraw the specified windows.
  - **settext:** Modify the caption/title of the specified windows.
  - **focus:** Set the focus to the specified window.
  - **disable:** Disable the specified window.
  - **enable:** Enable the specified window.
  - **toggledisable:** Toggles the specified windows between disabled and enabled state.
  - **+style:** Add style bits to the specified window. The value of **[Additional Parameters]** specifies the style bits to add.
  - **-style:** Remove style bits from the specified window. The value of **[Additional Parameters]** specifies the style bits to remove.
  - **+exstyle:** Add extended style bits to the specified window. The value of **[Additional Parameters]** specifies the extended style bits to add.
  - **-exstyle:** Remove extended style bits from the specified window. The value of **[Additional Parameters]** specifies the extended style bits to remove.
  - **postmsg:** Post a message to the specified window. The 3 values in **[Additional Parameters]** specifies the message parameters: Msg, wParam, lParam.

- **sendmsg**: Send a message to the specified window. The 3 values in **[Additional Parameters]** specifies the message parameters: Msg, wParam, lParam.
- **dlgclick**: Send a click command to the button inside a dialog-box. The value in **[Additional Parameters]** should specify the control ID of the button, or one of the following predefined buttons: yes, no, ok, cancel, retry, ignore, close, help.
- **dlgsettext**: Set the text to the specified control inside a dialog-box. The value in **[Additional Parameters]** should specify the control ID.
- **dlgsetfocus**: Set the focus to the specified control inside a dialog-box. The value in **[Additional Parameters]** should specify the control ID
- **[find]**: The method to find the window
  - **handle**: Finds the desired window by specifying the handle of the window in **[window to find]** parameter.
  - **id**: Finds the desired child window by specifying the id of the child window. Useful only for child windows.
  - **class**: Finds the desired window by specifying the class name of the window in **[window to find]** parameter.
  - **title**: Finds the desired window by specifying the exact title of the window in **[window to find]** parameter.
  - **stitle**: Finds the desired window by specifying the first few characters of the window in **[window to find]** parameter.
  - **ititle**: Finds the desired window by specifying a sequence of characters that exists in the window title.
  - **etitle**: Finds the desired window by specifying the last few characters of the window in **[window to find]** parameter.
  - **alltop**: All top windows.
  - **alltopnodesktop**: All top windows, except of desktop and tray windows.
  - **active**: The current active window.
  - **foreground**: The window in foreground.
  - **desktop**: The desktop window
  - **process**: Finds the desired window by specifying process ID (for example: /3412) or process name (for example: firefox.exe).
- **[window to find]**: Specifies the window by handle, class name, or title, according to the value of the previous parameter.

Examples:

```
win close title "calculator"
win hide stitle "calculat"
win hide ititle "culat"
win show class "SciCalc"
win activate title "my computer"
win flash title "calculator" 10 500
win max handle 0x00009c01
win redraw alltop
win trans ititle "internet explorer" 192
win setsize class "IEFrame" 50 50 400 300
win settopmost class "SciCalc" 1
win child class "Shell_TrayWnd" hide class "button"
win child class "Shell_TrayWnd" show class "button"
win child title "calculator" hide id 128
win -style title "my computer" 0x00C00000
win child title "my computer" +exstyle all 0x00400000
win sendmsg title "calculator" 0x10 0 0
win dlgclick class "#32770" cancel
win max process firefox.exe
```

```
win max process /1275
win togglehide class MozillaWindowClass
```

### cdrom [action] {drive:}

This command allows you to open and close the door of your CD-ROM drive. The **[action]** parameter can be 'open' or 'close'. If you don't specify the **{drive:}** parameter, the default CD-ROM will be used.

Examples:

```
cdrom open
cdrom close
cdrom open J:
cdrom close R:
```

### clipboard [Action] [Parameter]

Allows you to set a value into the clipboard and clear the clipboard.

[Action] may contain one of the following values:

- **set** - set the specified text into the clipboard.
- **readfile** - set the content of the specified text file into the clipboard.
- **clear** - clear the clipboard.
- **writefile** - write the content of the clipboard to a file. (text only)
- **writeufile** - write the content of the clipboard to a Unicode file. (text only)
- **addfile** - add the content of the clipboard to a file. (text only)
- **addufile** - add the content of the clipboard to a Unicode file. (text only)
- **saveimage** - Save the current image in the clipboard into a file.
- **copyimage** - Copy the content of the specified image file to the clipboard.
- **saveclp** - Save the current clipboard data into Windows .clp file.
- **loadclp** - Load Windows .clp file into the clipboard.

For copyimage and saveimage actions, the following file formats are supported: .bmp, .gif, .jpg, .png, .tiff - Only works on computers with GDI+ installed (On XP/2003/Vista, GDI+ is installed as a part of the operating system).

Examples:

```
clipboard set "123456"
clipboard set ~$folder.desktop$
clipboard set ~$folder.start_menu$
clipboard readfile "c:\temp\info1.txt"
clipboard addfile "c:\temp\clip1.txt"
clipboard clear
clipboard saveimage "c:\temp\clip01.png"
clipboard saveimage "c:\temp\clip01.gif"
clipboard copyimage "c:\temp\img01.gif"
clipboard saveclp "c:\temp\1.clp"
clipboard loadclp "c:\temp\1.clp"
```

### gac [Action] [Parameter]

Allows you to install an assembly in the Global Assembly Cache.

Examples:

```
gac install "C:\temp\MyAssembly\bin\Release\6\MyAssembly.dll"
```

### beep [Frequency] [Duration]



Plays a beep. The **[Frequency]** parameter specifies the frequency, in hertz. The **[Duration]** parameter specifies the duration of the sound in milliseconds. Under Windows 95/98/ME, the above parameters are ignored, and only a standard Windows beep is played.

Example:

```
beep 500 2000
```

### **stdbeep**

Plays the standard beep of Windows.

Example:

```
stdbeep
```

### **setvolume [device] [left-volume] [right-volume]**

Set the volume level of your sound device. The **[device]** parameter represents the device identifier. In most cases, you should use '0' value for this parameter. Under Windows Vista, you should specify 0 for default playback device or 1 for default recording device.

The **[left-volume]** and **[right-volume]** parameters should contain a value between 0 (silence) and 65535 (full volume) that represents the volume level for left and right channels.

Examples:

```
setvolume 0 0 0
```

```
setvolume 0 32768 32768
```

```
setvolume 0 0xffff 0
```

### **setsysvolume [volume] {Component} {Device Index}**

Set the sound volume for your system. Changing this value also affects the sound volume settings under 'Control Panel'. The **[volume]** parameter should contain a value between 0 (silence) and 65535 (full volume).

**{Component}** is an optional parameter that specifies one of the following sound components: master, waveout, synth, cd, microphone, phone, aux, line, headphones, wavein. The default value is 'master' (for master volume).

Under Windows Vista, the component name must be identical to the one of the device names appear in the sound window of Vista (Control Panel -> Sound). If the device name contains spaces, it must be in quotes. For Windows Vista, you can also specify 'default\_record' if you want to use the default recording device.

**{Device Index}** is an optional parameter that specifies the sound device index. This parameter is useful if you have more than one sound card. If you don't specify this parameter, the default sound card in Control Panel is used.

Examples:

```
setsysvolume 0x8000
```

```
setsysvolume 0
```

```
setsysvolume 25000 waveout
```

```
setsysvolume 25000 master 1
```

Vista Examples:

```
setsysvolume 0x8000
```

```
setsysvolume 0 "front mic"
```

```
setsysvolume 25000 speakers
```

```
setsysvolume 25000 "line in"
```

```
setsysvolume 25000 default_record
```

### **setsysvolume2 [left volume] [right volume] {Component} {Device Index}**

Similar to setsysvolume, but instead of setting the whole sound volume, setsysvolume2 set the right channel and left channel separately.

Examples:



```
setsysvolume2 10000 20000
setsysvolume2 30000 0
setsysvolume2 25000 15000 waveout
setsysvolume2 30000 0 master 1
```

### **changesysvolume [volume change] {Component} {Device Index}**

Increases or decreases the system volume. the **[volume change]** parameter specifies the number of volume units to increase or decrease (negative value to decrease, positive value to increase).

**{Component}** is an optional parameter that specifies one of the following sound components: master, waveout, synth, cd, microphone, phone, aux, line, headphones, wavein. The default value is 'master' (for master volume). Under Windows Vista, the component name must be identical to the one of the device names appear in the sound window of Vista (Control Panel -> Sound). If the device name contains spaces, it must be in quotes.

**{Device Index}** is an optional parameter that specifies the sound device index. This parameter is useful if you have more than one sound card. If you don't specify this parameter, the default sound card in Control Panel is used.

Examples:

```
changesysvolume 5000
changesysvolume -3000
changesysvolume 0x1000 waveout
changesysvolume 0x1000 master 1
```

Vista Examples:

```
changesysvolume 5000
changesysvolume -3000
changesysvolume 0x1000 "line in"
changesysvolume 0x1000 "speakers" 1
changesysvolume 0x1000 default_record
```

### **changesysvolume2 [left volume change] [right volume change] {Component} {Device Index}**

Similar to changesysvolume, but instead of changing the whole sound volume, changesysvolume2 changes the left channel and the right channel separately.

Examples:

```
changesysvolume2 1000 -1000
changesysvolume2 -3000 0
changesysvolume2 0 -5000 waveout
changesysvolume2 -3000 0 master 1
```

### **mutesysvolume [action] {Component} {Device Index}**

Mute/unmute the system volume.

In the **[action]** parameter, specify 1 for mute, 0 for unmute, or 2 for switching between mute and normal state.

The second parameter is optional, and you can specify one of the following sound components: master, waveout, synth, cd, microphone, phone, aux, line, headphones, wavein. The default value is 'master' (for master volume). Under Windows Vista, the component name must be identical to the one of the device names appear in the sound window of Vista (Control Panel -> Sound). If the device name contains spaces, it must be in quotes.

**{Device Index}** is an optional parameter that specifies the sound device index. This parameter is useful if you have more than one sound card. If you don't specify this parameter, the default sound card in Control Panel is used.

Examples:

```
mutesysvolume 1
mutesysvolume 0
mutesysvolume 2 waveout
mutesysvolume 1 master 1
```

Vista Examples:

```
mutesysvolume 1
mutesysvolume 0
mutesysvolume 2 "rear mic"
mutesysvolume 1 speakers 1
mutesysvolume 1 default_record
```

#### setappvolume [Process] [volume level] {Device Name/Index}

Set the volume level of specific application under Windows 7/Vista/2008.

In the **[process]** parameter, you can specify the .exe filename (For example: Firefox.exe, iexplore.exe, wmpayer.exe), or full path filename (For example: 'F:\Program Files\Internet Explorer\iexplore.exe') or the process ID (For example: /3426)

You can also specify 'focused' for this parameter, if you to want to set the volume of the focused application.

You can also specify 'SystemSounds' for this parameter, if you to want to set the 'System Sounds' volume.

The **[volume level]** parameter is a number between 0 and 1. For example, if you want to set the volume to 50%, you should set this parameter to 0.5

The **{Device Name/Index}** is optional parameter. If you don't specify it, the default output device is used. You can specify the device index as numeric value (0 = first device, 1 = second device, and so on) You also specify the full device name, like Speakers or Headphones. In order to find out what sound devices are available in your system, use the showsounddevices command.

Examples:

```
setappvolume iexplore.exe 0.2
setappvolume wmpayer.exe 0.75 Speakers
setappvolume /1275 0.25 1
setappvolume Firefox.exe 0.5
setappvolume focused 0.75
```

#### changeappvolume [Process] [volume level] {Device Name/Index}

Increase/decrease the volume level of specific application under Windows 7/Vista/2008.

In the **[process]** parameter, you can specify the .exe filename (For example: Firefox.exe, iexplore.exe, wmpayer.exe), or full path filename (For example: 'F:\Program Files\Internet Explorer\iexplore.exe') or the process ID (For example: /3426)

You can also specify 'focused' for this parameter, if you to want to change the volume of the focused application.

You can also specify 'SystemSounds' for this parameter, if you to want to set the 'System Sounds' volume.

The **[volume level]** parameter is a positive or negative number between 0 and 1. positive number increases the volume, negative number decreases the volume. For example, if you want to increase the volume from 20% (current volume) to 70%, you should set this parameter to 0.5

The **{Device Name/Index}** is optional parameter. If you don't specify it, the default output device is used. You can specify the device index as numeric value (0 = first device, 1 = second device, and so on) You also specify the full device name, like Speakers or Headphones. In order to find out what sound devices are available in your system, use the showsounddevices command.

Examples:

```
changeappvolume iexplore.exe -0.2
changeappvolume wmpayer.exe 0.55 Speakers
changeappvolume /1275 -0.25 1
changeappvolume Firefox.exe 0.5
```

#### muteappvolume [Process] [mute mode] {Device Name/Index}

Increase/decrease the volume level of specific application under Windows 7/Vista/2008.

In the **[process]** parameter, you can specify the .exe filename (For example: Firefox.exe, iexplore.exe, wmpplayer.exe), or full path filename (For example: 'F:\Program Files\Internet Explorer\iexplore.exe') or the process ID (For example: /3426)

You can also specify 'focused' for this parameter, if you to want to mute/unmute the volume of the focused application.

You can also specify 'SystemSounds' for this parameter, if you to want to set the 'System Sounds' volume.

The **[mute mode]** may contains one of the following values:

0 = Unmute

1 = Mute

2 = Switch between mute and unmute.

The **{Device Name/Index}** is optional parameter. If you don't specify it, the default output device is used. You can specify the device index as numeric value (0 = first device, 1 = second device, and so on) You also specify the full device name, like Speakers or Headphones. In order to find out what sound devices are available in your system, use the showsounddevices command.

Examples:

**muteappvolume iexplore.exe 0**

**muteappvolume wmpplayer.exe 1 Speakers**

**muteappvolume /1275 2 1**

**muteappvolume Firefox.exe 1**

#### **setdefaultsounddevice [Device Name] {Role}**

Set the default sound device on Windows 7/Vista/2008. The [Device Name] is the name of the device, as appeared in the sound devices list of windows, for example: Speakers, Line In, Microphone, and so on...

The {Role} parameter is optional and may contain one of the following values: 0 for Console (the default value), 1 for Multimedia, and 2 for Communications.

Examples:

**setdefaultsounddevice "Line In"**

**setdefaultsounddevice "Microphone" 2**

#### **setsubunitvolumedb [Device Name] [Subunit Name] [Decibel Value]**

Set the sound volume (in Decibels) of sound device subunit.

The **[Device Name]** is the name of the device, as appeared in the sound devices list of windows. In most cases the device name is 'Speakers'.

The **[Subunit Name]** is the name of a subunit, as appeared in the 'Level' tab of the device properties window.

Examples for subunit name: Phone Line, Microphone, Internal MIC, Line In, and so on...

Examples:

**setsubunitvolumedb "Speakers" "Line In" -25**

**setsubunitvolumedb "Speakers" "Microphone" -10**

#### **mutesubunitvolume [Device Name] [Subunit Name] [mute mode]**

Mute/unmute the sound volume of sound device subunit.

The **[Device Name]** is the name of the device, as appeared in the sound devices list of windows. In most cases the device name is 'Speakers'.

The **[Subunit Name]** is the name of a subunit, as appeared in the 'Level' tab of the device properties window.

Examples for subunit name: Phone Line, Microphone, Internal MIC, Line In, and so on...

The **[mute mode]** may contains one of the following values:

0 = Unmute  
1 = Mute  
2 = Switch between mute and unmute.  
Examples:  
**mutesubunitvolume "Speakers" "Line In" 0**  
**mutesubunitvolume "Speakers" "Microphone" 1**

#### **showsounddevices**

Displays a message box with the list of all sound devices on Windows Vista/7/2008.  
This command is useful for Windows Vista/7/2008, to easily find out which sound devices you can use on your system with the other sound volume commands.

#### **wait [milliseconds]**

Simply wait the specified number of milliseconds.  
Examples:  
**wait 1000**  
**wait 500**

#### **shellrefresh**

Initiate a general refresh for Windows Explorer. This refresh command can be useful if you make a change in the Registry related to the shell file types (For example, if you change the icon of .gif extension), and you want that your changes will take affect immediately.

#### **sysrefresh {Refresh Type}**

Initiate a general system refresh. You can use this command to refresh your system after changing your system settings in the Registry (e.g: changing desktop icons size, environment variables, and so on..) {Refresh Type} is optional. You can specify one of the following values: "environment" - for refreshing the environment variables, "policy" - for policy settings, "intl" for locale settings.

Examples:  
**sysrefresh**  
**sysrefresh environment**  
**sysrefresh policy**

#### **setcursor [X] [Y]**

Set the position of mouse cursor. The **[X]** and **[Y]** parameters specifies the desired cursor position.  
Example:  
**setcursor 100 50**

#### **setcursorwin [X] [Y]**

Set the position of mouse cursor, relative to the top-left corner of the active window. The **[X]** and **[Y]** parameters specifies the desired cursor position.  
Example:  
**setcursor 100 50**

#### **movecursor [X] [Y]**

move the mouse cursor. The **[X]** and **[Y]** parameters specifies the number of pixels to move. Example:  
**movecursor 10 10**

#### **dlg [Process Name] [Window Title] [Action] [Parameters]**

Allows you to interact with standard dialog-boxes and message-boxes of Windows. When a dialog-box is opened, you can use this command to "click" the ok/cancel/yes/no buttons, or fill the text-boxes in the dialog-box.

For example, the following commands fill the standard 'Run' dialog-box of Windows with 'calc.exe' process, and then click the 'OK' button. The '12298' is the control ID of the text-box in the Run dialog-box. You can find the control ID by using Spy++ of Microsoft, or by using my [WinExplorer](#) utility.

```
dlg "explorer.exe" "run" settext 12298 "calc.exe"  

dlg "explorer.exe" "run" click ok
```

The following command will choose the 'Yes' answer for any question dialog-box of Explorer process:

```
dlg "explorer.exe" "" click yes
```

The following command will choose the 'Cancel' answer for any question dialog-box of any process:

```
dlg "" "" click cancel
```

Parameters description:

- **[Process Name]**: Specifies the process that created the desired window. You can specify only the process name or the full path of the process. If this parameter is empty string(""), the command will be executed on any process.
- **[Window Title]**: Specifies the title of the window that you want the execute the action. If this parameter is empty string(""), the command will be executed on any window, regardless the window title.
- **[Action]**: You can specify one of the following options:
  - **click**: Click the specified button. You can specify one of the following predefined values (For standard Windows dialog-boxes only !): yes, no, ok, cancel, retry, ignore, close, help. You can also specify any control ID as numeric value.
  - **settext**: Set the text of the specified control. The first parameter of this action specifies the ID of the control, and the second parameter specifies the text.

#### **dlgany [Process Name] [Window Title] [Action] [Parameters]**

similar to dlg command, but also works on any window, even if it's not identified as a standard Windows dialog-box.

#### **returnval [Return Value]**

This command allows you to specify the return value for NirCmd. The return value can be used by external program that run NirCmd.

For example, in the following command, NirCmd returns 0x30 value if the user choose the 'Yes' answer:

```
qboxcom "Do you want to continue ?" "question" returnval 0x30
```

#### **sendkeypress [Keys Combination 1] [Keys Combination 2] [Keys Combination 3] ...**

Sends one or more key press combinations to the system. The operating system will behave exactly as the user really pressed the specified keys combination.

The **[Key Combination]** parameter specifies a single key press to send or a combination of a single key and

shift/ctrl/alt/Windows keys, delimited by '+' character.

The key in the **[Key Combination]** parameter can be specified as numeric [virtual key code](#) (For example: 0x2e for Delete key), or as one of the following predefined values: a - z and 0 - 9 (for alphanumeric keys), F1 - F24 (for Fxx keys), shift, ctrl, alt, enter, esc, leftshift, rightshift, leftctrl, rightctrl, leftmenu, rightmenu, spc (space), down, up, left, right, home, end, insert, delete, plus, comma, minus, period, lwin, rwin (Windows key), apps, pageup, pagedown, tab, multiply, add, subtract, separator, divide, backspace, pause, capslock, numlock, scroll, printscreen.

You can specify more than one **[Key Combination]** in a single sendkeypress command.

Here's some examples for using sendkeypress:

- Send ctrl+shift+esc to show the task manager of Windows:  
sendkeypress ctrl+shift+esc
- Send ctrl+shift+esc to show the task manager of Windows and then move to the next window with alt+tab:  
sendkeypress ctrl+shift+esc alt+tab
- Switch the capslock mode:  
sendkeypress capslock
- Print 'Nir Sofer' to the current text editor in focus (The shift key combination is used to get the capital letters):  
sendkeypress Shift+n i r spc shift+s o f e r
- Save the current screen image into the clipboard (Ctrl+PrintScreen):  
sendkeypress Ctrl+printscreen
- Send Ctrl+Shift+K and then Ctrl+Shift+A and then Alt+Shift+P:  
sendkeypress ctrl+shift+k ctrl+shift+a alt+shift+p

### sendkey [Key] [press | down | up]

Sends a keystroke to the system. The operating system will behave exactly as the user really pressed the specified keys.

The **[Key]** parameter specifies the key as [virtual key code](#) (For example: 0x2e for Delete key), or as one of the following predefined values: a - z and 0 - 9 (for alphanumeric keys), F1 - F24 (for Fxx keys), shift, ctrl, alt, enter, esc, leftshift, rightshift, leftctrl, rightctrl, leftmenu, rightmenu, spc (space), down, up, left, right, home, end, insert, delete, plus, comma, minus, period, lwin, rwin (Windows key), apps, pageup, pagedown, tab, multiply, add, subtract, separator, divide, backspace, pause, capslock, numlock, scroll, printscreen.

The second parameter specifies the key action: 'press', 'down', or 'up'. For each key down action, you should also add the appropriate key up action.

In the following script example, the system behaves like the user presses Shift+Ctrl+Esc (the task manager should appear):

```
sendkey shift down
sendkey ctrl down
sendkey esc down
sendkey shift up
sendkey ctrl up
sendkey esc up
```

More examples:

```
sendkey a press
sendkey f10 press
sendkey enter press
sendkey 0x2e press
```

**sendmouse [right | left | middle] [down | up | click | dblclick]**

**sendmouse [move] [x] [y]**

**sendmouse [wheel] [Wheel Value]**

Sends the specified mouse event to the system. The operating system will behave exactly as the user really made the specified mouse action.

Heres some example of sendmouse command:

- Sent a right click (For most applications, a context menu is opened):  
**sendmouse right click**
- Sent a double-click with the left mouse button:  
**sendmouse left dblclick**
- Press the left mouse button, move the mouse cursor 30 pixels left and 20 pixels down, and then release the button:  
**sendmouse left down**  
**sendmouse move -30 20**  
**sendmouse left up**
- Scroll the mouse wheel 10 units in standard wheel mouse. (On standard wheel mouse, the wheel value should be a multiple of 120).  
**sendmouse wheel 1200**

**convertimage [source filename] [destination filename]**

Convert image file from one format from another. The following file formats are supported: .bmp, .gif, .png, .jpg, .tiff

This command only works on Windows XP/2003/Vista (GDI+ is required)

Examples:

**convertimage "c:\temp\test1.gif" "c:\temp\test1.png"**  
**convertimage "c:\temp\test1.jpg" "c:\temp\test1.gif"**

**convertimages [source wildcard] [extension]**

Convert multiple image files from one format from another. The following file formats are supported: .bmp, .gif, .png, .jpg, .tiff

This command only works on Windows XP/2003/Vista (GDI+ is required)

Examples:

**convertimages "c:\temp\\*.gif" .png**  
**convertimages "c:\temp\\*.jpg" .gif**

**savescreenshot [filename] {x y width height}**

save the screenshot of current screen into the specified image filename. The following file formats are supported: .bmp, .gif, .png, .jpg, .tiff

You can also specify \*clipboard\* instead of real filename if you want to save the screenshot into the clipboard.

This command only works on Windows XP/2003/Vista (GDI+ is required)

You can also specify 4 optional parameters - x, y, width, and height, if you want to save only a part of the screen.

Examples:

```
savescreenshot "c:\temp\shot.png"  
savescreenshot "c:\temp\shot.png" 50 50 300 200  
savescreenshot *clipboard* 150 150 400 400
```

#### **savescreenshotfull [filename]**

similar to savescreenshot command, but on computer with multiple monitors, this command will take the screenshot of all monitors, as opposed to savescreenshot command, which takes the screenshot of the primary monitor. You can also specify \*clipboard\* instead of real filename if you want to save the screenshot into the clipboard.

#### **savescreenshotwin [filename]**

save the screenshot of current active window into the specified image filename. The following file formats are supported: .bmp, .gif, .png, .jpg, .tiff  
You can also specify \*clipboard\* instead of real filename if you want to save the screenshot into the clipboard. This command only works on Windows XP/2003/Vista (GDI+ is required)

Examples:

```
savescreenshotwin "c:\temp\win.png"  
savescreenshotwin *clipboard*
```

#### **shellcopy [source file or wildcard] [destination] {Flags}**

Copy one or more files from one folder to another, and display a progress dialog, exactly like copying files in Explorer.

In the Flags section, you can specify one or more of the following values:

- **yestoall**: Automatically answer 'Yes' to all questions.
- **noerrorui**: Don't display errors in user interface.
- **silent**: Silent mode - don't display progress information.
- **nosecattr**: Don't copy the security attributes of the file.

Examples:

```
shellcopy "c:\temp\*.*" "d:\temp"  
shellcopy "c:\temp\*.*" "d:\temp" yestoall noerrorui
```

#### **filldelete [source file or wildcard]**

fill the specified file or wildcard with zero-bytes, and then delete the files. This means that the content of the files deleted with this command cannot be recovered.

Examples:

```
filldelete "c:\temp\*.*"  
filldelete "c:\temp\myfile.dat"
```

#### **speak [type] [text/File name] {rate} {volume} { .wav Output File name } {Output Format}**

Speaks the contents of the text or file that you specify, by using the Speech library (SAPI) that comes with Windows XP and Windows Vista. If { .wav Output File name } and {Output Format} are specified, the speaking is saved into a .wav file instead of speaking on your sound device.

In the [type] parameter, you can specify one of the following values:



- **text**: The second parameter contains a simple text.
- **xml**: The second parameter contains a text with Speech xml data.
- **file**: The second parameter contains a text filename.

The {rate} parameter is optional parameter that you can specify the Speech rate, a value between -10 (very slow) and 10 (very fast). The {volume} parameter is optional parameter that you can specify the volume of the Speech, a value between 0 and 100.

if {wav Output Filename} contains a .wav filename, you should also specify one of the following output format strings in the {Output Format} parameter:

8kHz8BitMono, 8kHz8BitStereo, 8kHz16BitMono, 8kHz16BitStereo, 11kHz8BitMono, 11kHz8BitStereo, 11kHz16BitMono, 11kHz16BitStereo, 12kHz8BitMono, 12kHz8BitStereo, 12kHz16BitMono, 12kHz16BitStereo, 16kHz8BitMono, 16kHz8BitStereo, 16kHz16BitMono, 16kHz16BitStereo, 22kHz8BitMono, 22kHz8BitStereo, 22kHz16BitMono, 22kHz16BitStereo, 24kHz8BitMono, 24kHz8BitStereo, 24kHz16BitMono, 24kHz16BitStereo, 32kHz8BitMono, 32kHz8BitStereo, 32kHz16BitMono, 32kHz16BitStereo, 44kHz8BitMono, 44kHz8BitStereo, 44kHz16BitMono, 44kHz16BitStereo, 48kHz8BitMono, 48kHz8BitStereo, 48kHz16BitMono, 48kHz16BitStereo, TrueSpeech\_8kHz1BitMono, CCITT\_ALaw\_8kHzMono, CCITT\_ALaw\_8kHzStereo, CCITT\_ALaw\_11kHzMono, CCITT\_ALaw\_11kHzStereo, CCITT\_ALaw\_22kHzMono, CCITT\_ALaw\_22kHzStereo, CCITT\_ALaw\_44kHzMono, CCITT\_ALaw\_44kHzStereo, CCITT\_uLaw\_8kHzMono, CCITT\_uLaw\_8kHzStereo, CCITT\_uLaw\_11kHzMono, CCITT\_uLaw\_11kHzStereo, CCITT\_uLaw\_22kHzMono, CCITT\_uLaw\_22kHzStereo, CCITT\_uLaw\_44kHzMono, CCITT\_uLaw\_44kHzStereo, ADPCM\_8kHzMono, ADPCM\_8kHzStereo, ADPCM\_11kHzMono, ADPCM\_11kHzStereo, ADPCM\_22kHzMono, ADPCM\_22kHzStereo, ADPCM\_44kHzMono, ADPCM\_44kHzStereo, GSM610\_8kHzMono, GSM610\_11kHzMono, GSM610\_22kHzMono, GSM610\_44kHzMono

Examples:

```
speak text ~$clipboard$
speak text "Please visit the Web site of NirSoft at http://www.nirsoft.net" 2 80
speak file "c:\temp\speak1.txt"
speak file "c:\temp\speak1.txt" 0 100 "c:\temp\speak.wav" 48kHz16BitStereo
```

#### elevate [Program] {Command-Line Parameters}

For Windows Vista/7/2008 only: Run a program with administrator rights. When the [Program] contains one or more space characters, you must put it in quotes.

Examples:

```
elevate notepad.exe
elevate notepad.exe C:\Windows\System32\Drivers\etc\HOSTS
elevate "c:\program files\my software\abc.exe"
```

#### elevatecmd [NirCmd command]

For Windows Vista/7/2008 only: Run a NirCmd command with administrator rights.

Examples:

```
elevatecmd regsetval binary "HKLM\Software\Test" "Value1" "6a cb ab dd 80 86 95"
```

#### runas [Domain\User] [Password] [Process and Command-Line Parameters]

Run a process with the specified user name/password. The first parameter can be specified in one of the 3 formats:  
User name only, Domain\User, and user@domain

Examples:

**runas admin1 Password1 "F:\Program Files\Mozilla Firefox\firefox.exe"**  
**runas Domain01\user01 myp234Jk c:\temp\myprogram.exe**  
**runas user01@domain.com Nm1923a "c:\temp\myprogram.exe" -a -k**

#### **runassystem [Process and Command-Line Parameters]**

Run a process with a 'SYSTEM' user. This command only works on Windows 7/8/2008/Vista. It also requires full administrator rights, so you should run it with elevatcmd command.

Example:

**elevatcmd runassystem c:\windows\regedit.exe**

#### **runinteractive [Process and Command-Line Parameters]**

Allows you to run a process that accesses the GUI from a Windows service.

Example:

**runinteractive c:\windows\system32\taskmgr.exe**

#### **runinteractivcmd [NirCmd command]**

Allows you to run a command that accesses the GUI from a Windows service.

Example:

**runinteractivcmd savescreenshot "c:\temp\screenshot1.png"**

#### **trayballoon [Title] [Balloon Text] [Icon File] [Timeout]**

Displays a tray balloon (For Windows XP and above). Be aware that when using this command, NirCmd will continue running until the balloon is closed or disappeared.

The [Title] parameter is the title of the balloon and the [Balloon Text] parameter specifies the text to display inside the balloon. In the [Icon File] parameter, you can specify .ico file (e.g.: c:\icons\myicon.ico) or .exe/.dll file with icon resource index or number (e.g.: shell32.dll,-154)

The [Timeout] parameter specifies the number of milliseconds to wait before the balloon is disappeared.

Examples:

**trayballoon "Hello" "This is a test..." "shell32.dll,22" 15000**  
**trayballoon "Hello" "This is the text that will be appear inside the balloon !" "shell32.dll,-154" 10000**  
**trayballoon "Hello" "This is the text that will be appear inside the balloon !" "c:\temp\icon1.ico" 10000**

#### **setconsolemode [mode]**

Set the console display mode. Relevant only for nircmdc.exe

In the [mode] parameter, you can specify 1 for full screen mode or 2 for console window mode.

#### **setconsolecolor [forecolor] [backcolor]**

Relevant only for nircmdc.exe

Set the current background and foreground color for the text in the console window. This command won't change the color of the existing text, but all text sent to stdout (with consolewrite command) after this command will be displayed in the specified color.

In the forecolor/backcolor parameters, you can specify any EGA color number between 0 and 15:

- 0 - black
- 1 - blue
- 2 - green
- 3 - cyan
- 4 - red
- 5 - magenta
- 6 - brown
- 7 - white
- 8 - dark gray
- 9 - bright blue
- 10 - bright green
- 11 - bright cyan
- 12 - bright red
- 13 - bright magenta
- 14 - bright yellow
- 15 - bright white

Examples:

**setconsolecolor 10 0**

**setconsolecolor 1 15**

#### **consolewrite [text]**

Sends the specified text to the standard output (stdout).

#### **debugwrite [text]**

Writes the specified text into the debug output. You can view the text sent with this command in the [DebugView](#) utility of SysInternals.

#### **mediaplay [play time] [audio file]**

Plays the specified audio file (.mp3, .wav, and so on) for the specified number of milliseconds.

Examples:

**mediaplay 10000 "c:\temp\1.mp3"**

**mediaplay 180000 "c:\my files\1.wav"**