

## KAK - The DevCon command-line utility functions as an alternative to Device Manager

The DevCon utility is a command-line utility that acts as an alternative to Device Manager. Using DevCon, you can enable, disable, restart, update, remove, and query individual devices or groups of devices. DevCon also provides information that is relevant to the driver developer and is not available in Device Manager.

You can use DevCon with Microsoft Windows 2000, Windows XP, and Windows Server 2003. You cannot use DevCon with Windows 95, Windows 98, or Windows Millennium Edition.

DevCon is not redistributable. It is provided for use as a debugging and development tool. You can freely modify DevCon for private use. The sample demonstrates how to use the SetupAPI and CfgMgr32 functions together effectively to enumerate devices and perform device operations. The following file is available for download from the Microsoft Download Center: Release Date: Jan-29-2003

For more information about how to download Microsoft support files, click the following article number to view the article in the Microsoft Knowledge Base: [119591](#) How to obtain Microsoft support files from online services Microsoft scanned this file for viruses. Microsoft used the most current virus-detection software that was available on the date that the file was posted. The file is stored on security-enhanced servers that help prevent any unauthorized changes to the file.

The DevCon.exe file contains the following files:

File	Description
I386\DevCon.exe	32-bit DevCon tool binary. This will not function completely on 64-bit Windows.
Ia64\DevCon.exe	64-bit DevCon tool binary.

**Note** The source code for DevCon is also available in the Windows DDK (which is available from <http://www.microsoft.com/whdc/devtools/ddk/default.mspx>) under *DDK root\Src\Setup\Devcon*, along with documentation.

### Using DevCon

DevCon is a command-line utility with built-in documentation. If you run the **devcon help** command, the following list of commands and descriptions appears. The **devcon help** command will give more detailed help on any command. With some of these commands, you can specify a remote target computer. These commands work if you are using the 32-bit version of DevCon on WOW64.

```
Device Console Help:
devcon.exe [-r] [-m:\\<machine>] <command> [<arg>...]
-r if specified will reboot machine after command is complete, if needed.
<machine> is name of target machine.
<command> is command to perform (see below).
<arg>... is one or more arguments if required by command.
```

For help on a specific command, type: devcon.exe help <command>

```
classfilter      Allows modification of class filters.
classes          List all device setup classes.
disable          Disable devices that match the specific hardware or instance ID.
driverfiles      List driver files installed for devices.
drivernodes      Lists all the driver nodes of devices.
enable           Enable devices that match the specific hardware or instance ID.
find             Find devices that match the specific hardware or instance ID.
findall         Find devices including those that are not present.
help             Display this information.
hwids            Lists hardware ID's of devices.
install          Manually install a device.
listclass        List all devices for a setup class.
reboot           Reboot local machine.
remove           Remove devices that match the specific hardware or instance ID.
rescan           Scan for new hardware.
resources        Lists hardware resources of devices.
restart          Restart devices that match the specific hardware or nstance ID.
stack            Lists expected driver stack of devices.
status           List running status of devices.
update           Manually update a device.
UpdateNI         Manually update a device without user prompt
SetHwID          Adds, deletes, and changes the order of hardware IDs of root-enumerated devices.
```

### Example DevCon commands

```
devcon -m:\\test find pci\*
```

Lists all known PCI devices on the computer test. (By using -m, you can specify a target computer. You must use Interprocess communication (IPC) to access the computer.)

```
devcon -r install %WINDIR%\Inf\Netloop.inf *MSLOOP
```

Installs a new instance of the Microsoft loopback adaptor. This creates a new root-enumerated device node with which you can install a "virtual device," such as the loopback adaptor. This command also restarts the computer silently if a restart is required.

```
devcon classes
```

Lists all known setup classes. The output contains the short nonlocalized name (for example, "USB") and the descriptive name (for example, "Universal Serial Bus controllers").

```
devcon classfilter upper !filter1 !filter2
```

Deletes the two specified filters.

```
devcon classfilter lower !badfilter +goodfilter
```

Replaces the "badfilter" with the "goodfilter".

```
devcon driverfiles =ports
```

Lists files that are associated with each device in the ports setup class.

```
devcon disable *MSLOOP
```

Disables all devices that have a hardware ID that ends in "MSLOOP" (including "\*MSLOOP").

```
devcon drivernodes @ROOT\PCI_HAL\PNP0A03
```

Lists all compatible drivers for the device ROOT\PCI\_HAL\PNP0A03. This can be used to determine why an integral device information (.inf) file was chosen, instead of a third-party .inf file.

```
devcon enable '*MSLOOP
```

Enables all devices that have a hardware ID of "\*MSLOOP". The single quotation mark indicates that the hardware ID must be taken literally (in other words, the asterisk ["\*"] actually is an asterisk; it is not a wildcard character).

```
devcon find *
```

Lists device instances of all devices that are present on the local computer.

```
devcon find pci\*
```

Lists all known peripheral component interconnect (PCI) devices that are on the local computer (this command assumes that a device is PCI if it has a hardware ID that is prefixed by "PCI\").

```
devcon find =ports *pnp*
```

Lists devices that are present that are a member of the ports setup class and that contain "PNP" in their hardware ID.

```
devcon find =ports @root\*
```

Lists devices that are present that are a member of the ports setup class and that are in the "root" branch of the enum tree (the instance ID is prefixed by "root\"). Note that you should not make any programmatic assumption about how an instance ID is formatted. To determine root devices, you can look at device status bits. This feature is included in DevCon to aid in debugging.

```
devcon findall =ports
```

Lists "nonpresent" devices and devices that are present for the ports class. This includes devices that have been removed, devices that have been moved from one slot to another, and, in some cases, devices that have been enumerated differently due to a BIOS change.

```
devcon listclass usb 1394
```

Lists all devices that are present for each class named (in this case, USB and 1394).

```
devcon remove @usb\*
```

Removes all USB devices. Devices that are removed are listed with their removal status.

```
devcon rescan
```

Rescans for new Plug and Play devices.

```
devcon resources =ports
```

Lists the resources that are used by all devices in the ports setup class.

```
devcon restart =net @'ROOT\*MSLOOP\0000
```

Restarts the loopback adaptor ROOT\\*MSLOOP\0000. The single quotation mark in the command indicates that the instance ID must be taken literally.

```
devcon hwids=mouse
```

Lists all hardware IDs of mouse class devices on the system.

```
devcon sethwid @ROOT\LEGACY_BEEP\0000 := beep
```

Assign the hardware ID, beep, to the legacy beep device.

```
devcon stack =ports
```

Lists the expected driver stack for the device. This includes device and class upper/lower filters, and the controlling service.

```
devcon status @pci\*
```

Lists the status of each device present that has an instance ID that begins with "pci\".

```
devcon status @ACPI\PNP0501\1
```

Lists the status of a specific device instance, in this case an Advanced Configuration and Power Interface (ACPI)-enumerated serial port.

```
devcon status @root\rdp_mou\0000
```

Lists the status of the Microsoft Terminal Server or Terminal Services mouse driver.

```
devcon status *PNP05*
```

Lists the status of all COM ports.

```
devcon update mydev.inf *pnp0501
```

Updates all devices that exactly match the hardware ID \*pnp0501 to use the best driver in Mydev.inf that is associated with the hardware ID \*pnp0501.

### Note:

This update forces all devices to use the driver in Mydev.inf, even if there is a better match already on the system. This is useful when you want to install new versions of drivers during development before you obtain a signature. The update affects only the devices that match the specified hardware ID, and does not affect the child devices. If the specified .inf file is unsigned, Windows may display a dialog box that prompts you to confirm whether the driver should be installed. If a restart is required, this is reported and DevCon returns a level 1 error. If you specify -r, this causes a restart to occur automatically if one is required.

### Notes

- DevCon will return an error level for use in scripts:
  - "0" indicates a success.
  - "1" indicates that a restart is required.
  - "2" indicates a failure.
  - "3" indicates a syntax error.
- If you specify -r and a restart is required, the restart occurs without warning after all devices have been processed.
- If you specify -m:\\computer and the command will not work for a remote computer, an error is reported.
-

DevCon allows wildcards in instance IDs for interactive convenience. Do not assume anything about the format of an instance ID from computer to computer and from operating system version to operating system version.