# DXLab Launcher Online Help

The DXLab Launcher  enables you to
- start or stop a group of DXLab applications and up to four other applications with a single mouse click
- minimize or restore the main windows of all running DXLab applications with a single mouse click
- start, minimize, restore, or stop individual DXLab applications
- view at a glance the state of all DXLab applications
- compare the versions of DXLab applications currently installed with the versions currently available, and visit a site from which new versions can be downloaded
- save all DXLab applications Windows Registry settings to a workspace (a sub-folder in the Launcher's Workspaces folder)
- restore all DXLab applications Windows Registry settings from a workspace (a sub-folder in the Launcher's Workspaces folder)
- save a DXLab application's Windows Registry settings to a specified file
- restore a DXLab application's Windows Registry settings from a specified file
- delete a DXLab application's Windows Registry settings
- direct all applications to place their windows on the primary monitor

**Prerequisites**

**Downloading and Installation**

**Configuring the Launcher**

**Controlling DXLab Applications**

**Saving and Restoring Registry Settings**

# DXLab Launcher Prerequisites

To use the DXLab Launcher, you need
- a PC running Windows 95, Windows 98, Windows 2000, or Windows NT, ideally
    - 133 mHz Pentium or better
    - 64 MB RAM or better
- an SVGA display or better

If you're only running 1 or 2 DXLab applications, you probably don't need the Launcher. The more DXLab applications you run, the more convenient you'll find the Launcher as a central means of starting or stopping them all with a single mouse click. The more DXLab applications you run, however, the more memory may be required for satisfactory performance.

# DXLab Launcher Download and Installation

All DXLab applications are produced using a process that generates frequent releases, referred to as **development releases**. Every few months, a stable development release is used to create a **full release** containing all software and documentation components. Development releases contain only those components that have changed since the most recent full release. Thus installing DXLab Launcher on a PC for the first time is a two-step process:

1. install the most recent full release
2. install the most recent development release

The instructions below describe how to install the most recent full release. When you've completed these steps, check http://www.qsl.net/dxlab/download.htm for access to the latest development release and instructions for installing it.

| Step | Directions |
|---|---|
| **1** | Create the application folder in which the DXLab Launcher will reside, such as: `C:\Program Files\DXLabLauncher` |
| **2** | In the application folder, create an installation subfolder in which the downloaded and extracted files will reside, such as: `C:\Program Files\DXLabLauncher\Install` |
| **3** | Go to http://www.qsl.net/dxlab/Launcher/DXLabLauncher115Archive.exe to download DXLabLauncher115Archive.exe, a 1,5 MB self-extracting executable file that contains the DXLab Launcher and its associated files. When prompted, direct your browser to store this file into the installation subfolder you created in step 2. |
| **4** | Run DXLabLauncher115Archive.exe - in its WinZip Self-Extractor dialog box, direct it to place the unzipped files into the installation subfolder you created in step 2, and click the Unzip button. After extraction completes, click the Close button. The following files should now be present in the installation subfolder:<br>• DXLabLauncher115Archive.exe<br>• DXLab1.cab<br>• DXLab2.cab<br>• setup.exe<br>• Setup.lst |
| **5** | To install the DXLab Launcher, run the setup.exe program in the installation subfolder. Ignoring the setup program's request to close all running applications may result in error messages during the installation process, and possibly a faulty installation.<br><br>After copying several system files, the setup program may ask that you reboot your PC before continuing with the setup. If, after rebooting, your PC does not run setup.exe on its own, direct it to do so.<br><br>The setup program may report that the files being installed are older than files already installed on your system, and ask whether you want to over-write the existing newer files with the older files -- you should decline. |
| **6** | To execute the DXLab Launcher, run the program `DXLabLauncher.exe` in the application folder. |

**7**          After the DXLab Launcher is installed and you've verified that it works, you may delete the installation subfolder and the files it contains.

**8**          Check www.qsl.net/dxlab/download.htm to see if a development release is available

You can uninstall DXLab Launcher by running the Add/Remove Programs applet on the Windows control panel.

If you have questions or suggestions, please send them via email to the DXLab reflector at dxlab@yahoogroups.com.

# DXLab Launcher Configuration

Configuring the DXLab launcher requires that you
1.  Specify a pathname for each DXLab application you are using
2.  Specify the pathname of the log file to be opened by DXKeeper after startup
3.  Specify pathnames for non-DXLab applications to be started when you start a group of DXLab applications
4.  Specify Main Window Captions for non-DXLab applications to enable their termination by the Launcher
5.  Indicate which applications are to be started or stopped when you click the Main window's Start or Terminate buttons respectively
6.  Indicate whether the specified applications should be automatically started when the Launcher is started
7.  Indicate whether or not you want the Launcher's main window to always be "on top"
8.  Indicate whether the Launcher's Main window should automatically minimize after the specified applications are started
9.  Indicate whether the Launcher should check for new versions on startup
10. Indicate whether the Launcher should automatically shut down after you click the Terminate button

All of these actions can be accomplished in the Launcher's **Configuration** window; to display this window, click the Config button in the Launcher's Main window. The **Configuration** window contains two panels: the **DXLab Applications** panel, and the **Other Applications** panel. Each of these panels provides a row of controls for each application.

**Specifying a DXLab application's pathname**
To specify a DXLab application's pathname, locate the application's row in the **DXLab Applications** panel, and click the **Sel** button in this row. The Launcher will display an **Open** dialog box that makes it easy to select the application's executable file, such as
```
C:\Program Files\DXLab Suite\WinWarbler\WinWarbler.exe
```
or
```
D:\Program Files\SpotCollector\SpotCollector180.exe
```

**Specifying a log file pathname**
To specify the pathname of a log file to be opened by DXKeeper after startup, click the **Sel** button in the **DXLab Application** panel's **Startup Parameters** sub-panel, and choose the desired log file. If no log file pathname is specified, DXKeeper will open log in use during the previous operating session.

**Specifying a non-DXLab application's pathname**
The Launcher allows you to specify four non-DXLab applications that can be started when you start a group of DXLab applications; initially, these are depicted in the **Other Applications** panel as *Application 1* through *Application 4*. To specify the pathname of such an application, click the **Other Applications** panel's **Show Paths** button, click the **Sel** button associated with an unused **Program Path** textbox, and use the **Open** dialog box to select the application's executable file, e.g.
```
C:\Program Files\DX Atlas\DxAtlas.exe
```

When you select an executable file, the Launcher updates the **Application** column with the executable file's simple name. Using the preceding example, the entry in the **Application** column would be *DXAtlas*.

**Specifying a non-DXLab application's Main Window Caption**
If you have specified the caption of a non-DXLab application's Main Window, then clicking the Group Terminate button will terminate that non-DXLab application if its **Enable** box is checked n the **Other Applications** panel, or if the **CTRL** key was depressed when you clicked the Group Terminate button. To specify the Main Window caption of such an application, click the **Other Applications** panel's **Show Captions** button, and enter the caption in the application's **Main Window Caption** textbox. Note that the specified caption must exactly match the caption shown on the Windows Task Manager's Applications tab, including upper/lower case characters, spaces, and punctuation. Non-DXLab applications whose Main Window Caption changes over time cannot be terminated via the Group Terminate button.

**Indicating which applications are to be started or terminated as a group**
The Launcher's Main window lets you start or stop DXLab applications individually. It also allows you to start or stop a group of applications with a single mouse click. Each member of this group is designated by checking the associated **Enable** box in the **DXLab Applications** and **Other Applications** panels. If the **CTRL** key is depressed when the Group Terminate button is clicked, applications are terminated whether or not their associated Enable box is checked.

**Automatically starting the specified applications**
If you want the Launcher to automatically start the specified operations without waiting for you to click the Start button, then check  the **Configuration window**'s **Auto Start** box.

**Keeping the Launcher's Main window visible at all times**
If you want the Launcher's Main window to be visible even when another window is placed atop it, then check the **Configuration window**'s **Always On Top** box.

**Automatically minimizing the Launcher's Main window after starting the specified applications**
If you want the Launcher's Main window to automatically minimize after the specified applications are started (either by your clicking the **Start** button or by having the **Auto Start** box checked), then check the **Configuration window**'s **Minimize After Start** box.

**Automatically checking for new versions on startup**
If you want the Launcher to update the list of available versions and highlight any that are newer than those currently installed, then check the **Check for New on Start** box. You can check for new versions at any time by clicking the **New ?** button.

**Automatically shutting down the Launcher after terminating the specified applications**

If you want the Launcher  to automatically minimize shut down after you click the **Terminate** button and all applications have terminated, then check the **Configuration window**'s **Shutdown After Terminate** box.

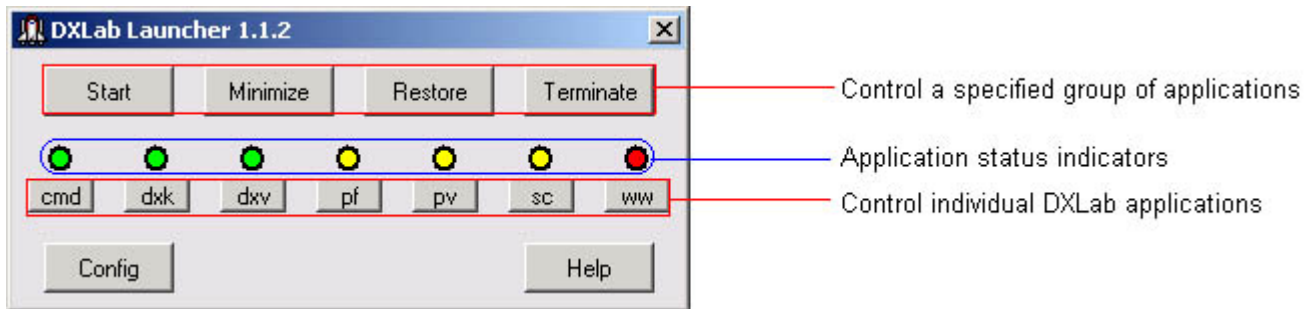**Resetting application windows to the primary monitor**
The Configuration window's **Reset** button directs all running DXLab applications to arrange their windows on the primary monitor. This operation is useful if you have switched from a dual-monitor configuration to a single-monitor configuration and need to regain access to windows previously resident on the second monitor.

When checked, the Configuration window's Log Debugging Information box produces a file named errorlog.txt that captures diagnostic information.

**Other Settings**
- Use Dual Monitors: when checked, windows that resided on the secondary monitor during the previous session will be restored to the secondary monitor on startup
- Log Debugging Info: when checked, debugging information will be recorded in the file errorlog.txt
- Browser pathname: lets you specify the pathname of the HTML browser that's used to display online documentation; click the associated **Sel** button to display a file selector window that lets you choose the desired browser.

# DXLab Launcher: Controlling Applications



As shown in the annotated screen shot above, the DXLab Launcher's Main window provides three primary groups of controls:

- A set of seven **Application Status Indicators** the indicate the state of each member of the DXLab Suite
- A set of seven **Individual Application Control Buttons** that enable you to change the state of each DXLab application
- A set of four **Group Control Buttons** that enable you to control multiple applications simultaneously

The Main window also provides buttons that display the Launcher's Configuration window and provide access to its online documentation.

The Configuration window's **Reset** button directs all running DXLab applications to arrange their windows on the primary monitor. This operation is useful if you have switched from a dual-monitor configuration to a single-monitor configuration and need to regain access to windows previously resident on the second monitor.

**Application Status Indicators**
Each application status indicator's color reveals the state of the DXLab application associated with the button below it:

| Indicator Color | Current State |
|---|---|
| red | not running |
| cyan | starting |
| green | running, not minimized |
| yellow | running, minimized |
| white | terminating |

**Individual Application Control Buttons**
Each of these buttons controls one of the seven DXLab applications:

| Caption | DXLab Application |
|---|---|
| cmd | Commander |
| dxk | DXKeeper |
| dxv | DXView |
| pf | Pathfinder |
| pv | PropView |
| sc | SpotCollector |
| ww | WinWarbler |

When you click one of these buttons, the Launcher's action is determined by two factors:
- the application's current state
- whether or not you depressed the **Ctrl** key while clicking the button

| Current Indicator Color | Current State | Action | New State | New Indicator Color |
|---|---|---|---|---|
| red | not running | user clicks **Individual Control Button** | starting | cyan |
| cyan | starting | application loads and runs | running, not minimized | green |
| green | running, not minimized | user clicks **Individual Control Button** | running, minimized | yellow |
| green | running, not minimized | user **Ctrl**-clicks **Individual Control Button** | terminating | white |
| yellow | running, minimized | user clicks **Individual Control Button** | running, not minimized | green |
| yellow | running, minimized | user **Ctrl**-clicks **Individual Control Button** | terminating | white |
| white | terminating | application terminates | not running | red |

Clicking an **Individual Control Button** starts a DXLab application that isn't running. If an application is running, clicking its **Individual Control Button** alternates between hiding all of its windows and restoring its Main window. user **Ctrl**-clicks **Individual Control Button**. Clicking a running application's **Individual Control Button** terminates that application.

If the Launcher's Main window is activated, allowing the mouse cursor to hover over an **Individual Control Button** will produce a popup message describing the action that would be taken if the button were to be clicked or **Ctrl**-clicked.

If DXKeeper is running and you minimize its windows by clicking its **Individual Control Button**, it remembers whether or not its Capture window was already minimized. If not, the DXKeeper will restore both its Main and Capture windows when you again click its **Individual Control Button.**

If DXView is running and you minimize its windows by clicking its **Individual Control Button**, it remembers whether or not its World Map window was already minimized. If not, the DXView will restore both its Main and World Map windows when you again click its **Individual Control Button.**

You can optionally specify the pathname of a log file to be opened by DXKeeper on startup.

**Group Control Buttons**

| Caption | Function |
|---|---|
| Start | • start all applications whose Enabled box is checked |
| Minimize | • hide all windows of all running DXLab applications |
| Restore | • restore the Main windows of all running DXLab applications |
| Terminate | • if the **CTRL** key is not depressed, terminate all applications whose Enabled box is checked<br>• if the **CTRL** key is depressed, terminate all applications whether or not their Enabled boxes are checked<br>• non-DXLab applications are only terminated if their Main Window Caption is specified (whether or not **CTRL** is depressed) |

If DXKeeper is running and you click the **Minimize Button**, it remembers whether or not its Capture window was already minimized. If not, the DXKeeper will restore both its Main and Capture windows when you click the **Restore Button.**

If DXView is running and you click the **Minimize Button**, it remembers whether or not its World Map window was already minimized. If not, the DXView will restore both its Main and World Map windows when you click the **Restore Button.**

You can optionally specify the pathname of a log file to be opened by DXKeeper on startup.

**Displaying the Launcher's Configuration window**
To display the Configuration window, click the Main window's **Config** button.

# DXLab Launcher: Saving and Restoring Registry Settings

In conformance with Windows programming standards, DXLab applications store their settings in the Windows Registry. Such settings persist between operating sessions, and also survive the installation of both full releases and development releases. There are several reasons to save a DXLab's registry settings to a file:

1. for archival storage, to be restored in the event of hardware or operating system failure
2. to copy settings from one PC to another
3. to establish multiple "operating configurations" and rapidly switch between them

**Saving and restoring DXLab application settings with workspaces**
The Launcher lets you save the settings of all specified DXLab applications into a *workspace* - a folder in the Launcher's `Configuration` folder; each application's settings are stored within the workspace folder in an appropriately named file whose suffix is `.reg`. You can *activate* a workspace: restore the registry settings from each `.reg` file contained in that workspace. The Launcher's Workspace window displays a list of available workspaces; letting the mouse cursor hover over a workspace in this list will display a popup window showing the names of DXLab applications for which settings are stored in that workspace.  To display the Workspace window, click the Configuration window's **Workspaces** button; the following functions are available:

| | |
|---|---|
| **New** | prompts you to specify a workspace name, and then creates the workspace<br>• since workspaces are folders, workspace names must conform to the rules for folder names |
| **Activate** | restores registry settings from each .reg file contained in the selected workspace<br>• most DXLab applications only check their registry settings on startup<br>• its a good idea to terminate all running DXLab applications except the Launcher before invoking this function; otherwise, currently running DXLab applications could overwrite the restored registry settings when you later terminate those applications |
| **Update** | saves the registry settings of all specified DXLab applications into `.reg` files in the selected workspace, over-writing any existing files in that workspace<br>• settings are saved for each DXLab application with a valid Program Path in the Configuration window's DXLab Applications panel |
| **Delete** | deletes the selected workspace |

If no workspace is selected, the **Activate**, **Update**, and **Delete** buttons will be disabled (grayed out); to select a workspace, click on its entry in the Workspace List.

For the purpose of backup and recovery, its recommended that you specify pathnames for all of your DXLab applications in the Configuration window's DXLab Applications panel; create a New workspace, and then Update this workspace. Backing up the Launcher's Workspaces folder (and its sub-folders) will make it easy to later recover, should your PC subsequently suffer hardware or software problems, or should you wish to install DXLab on another PC and have it use identical settings. **Important note**: if using Windows Explorer to navigate the Launcher's Workspaces folder and its sub-folders, do not double click on a `.reg` file unless you want that file's settings immediately loaded into the Windows registry!

To establish multiple operating configurations,
1. setup your DXLab applications for the first configuration
2. create a new workspace with an appropriate name, and select the newly-created workspace
3. click the Workspace window's **Update** button
4. if more configurations remain, setup your DXLab applications for the next configuration, and proceed to step 2

To switch from one operating configuration to another
1. terminate all running DXLab applications except the Launcher; the Launcher's Terminate button is a convenient way to do this
2. select the workspace representing the desired configuration
3. click the Workspace window's **Activate** button

**Saving, clearing, and restoring Windows Registry settings for individual DXLab applications**
The **Registry Settings** panel in the DXLab Launcher's Configuration window provides buttons that enable you to save a DXLab application's registry settings to a file of your choice, or load an application's registry settings from a file of your choice; the Windows standard `.reg` extension is used in both cases. When invoking either the **Save** or **Load** function, its recommended that the associated DXLab application not be running.

**Important note**: If you save a DXLab application's registry settings to a file and later wish to visually examine the contents of that file, do not invoke the Windows Explorer and double-click on the file. Rather than display the file's contents in Notepad, as happens when you double-click on a *.txt* file, double-clicking on a `.reg` file updates the Windows registry with the contents of the file -- just as if you'd performed the Launcher's **Load** function. To view the contents of a `.reg` file, start Notepad (or the editor of your choice) manually, and then **File:Open** the `.reg` file.

The **Registry Settings** panel also enables you to delete all of a DXLab application's Windows Registry settings; to do so, depress the CTRL key while clicking the associated **Ld** button. Before invoking this function, saving the application's registry settings to a file by clicking the **Sv** button is recommended.