

Launcher 2.3.2

Overview.....	2
DXLab Prerequisites	3
Download and Installation	4
Configuration	10
Controlling Applications.....	14
Installing, Upgrading, and Restoring DXLab Applications	20
Using Workspaces to Save and Load Windows Registry Settings	22

Overview

The DXLab Launcher enables you to

- determine whether new DXLab application upgrades or database upgrades are available
- install new DXLab applications, and upgrade or restore already installed DXLab applications
 - determine which DXLab applications are installed and which are uninstalled
 - install a DXLab application from a designated distribution web site or a local folder
 - compare the versions of installed DXLab applications with the versions currently available
 - upgrade a currently-installed application from a designated distribution web site or a local folder
 - restore a currently-installed application from a designated distribution web site or a local folder
- control the execution of DXLab and non-DXLab applications
 - start or stop a group of DXLab applications and up to sixteen 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
 - start individual non-DXLab applications
- manage the Windows registry settings of your DXLab applications
 - 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

DXLab Prerequisites

To run a DXLab application, you'll need

- a PC running Windows 98, Windows 2000, Windows NT, Windows XP, Windows Vista, Windows 7, or Windows 8, ideally
 - 1 GHz Pentium or better
 - 500 mb RAM or better
- an SVGA display (resolution 800 x 600) or better

To run multiple DXLab applications, you'll need a faster processor and more RAM. Windows 98's memory restrictions will not enable it to support the entire DXLab Suite.

Download and Installation

If you're installing the Launcher onto a PC running Vista, first login to the account named Administrator.

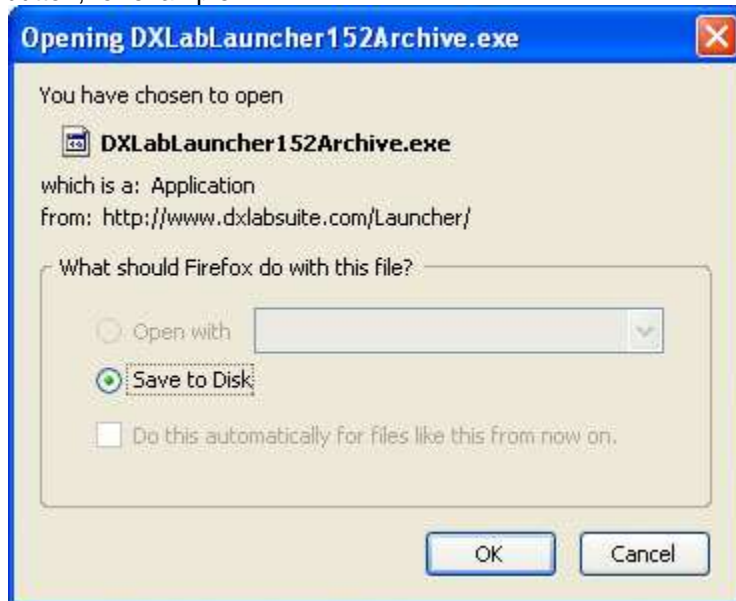
To install the DXLab Launcher, navigate to <http://www.dxlabsuite.com/download.htm> and click its **Install Launcher** button. When the following window appears, click its **Run** button:



A download progress window like this one will appear while the file is being downloaded:



If on clicking the **Install Launcher** button your web browser displays a window without a **Run** button, for example:

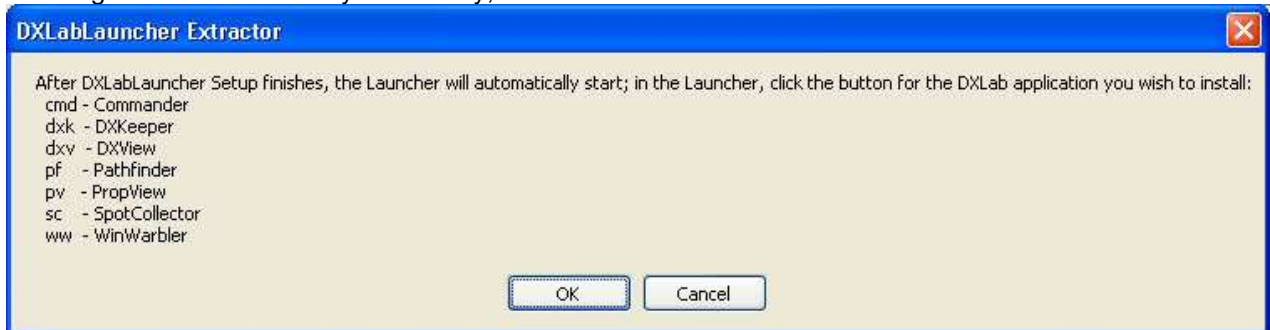


then create a new folder named **c:\DXLabLauncher**, click the window's **OK** button, and save the downloaded DXLabLauncher Archive file into the folder **c:\DXLabLauncher**. When the download is complete, use Windows Explorer to open **c:\DXLabLauncher**, and run the DXLabLauncher Archive file by double-clicking it.

If this window or one like it should appear, click its **Run** button:



The next window informs you that the **DXLabLauncher Setup** program will run, after which the Launcher itself will start. The abbreviated captions on the Launcher's buttons are expanded here so you'll know which button to click to install the desired DXLab application. Before proceeding, it's best to terminate any other applications running on your PC; this prevents conflicts when installing shared files. When you're ready, click the **OK** button:

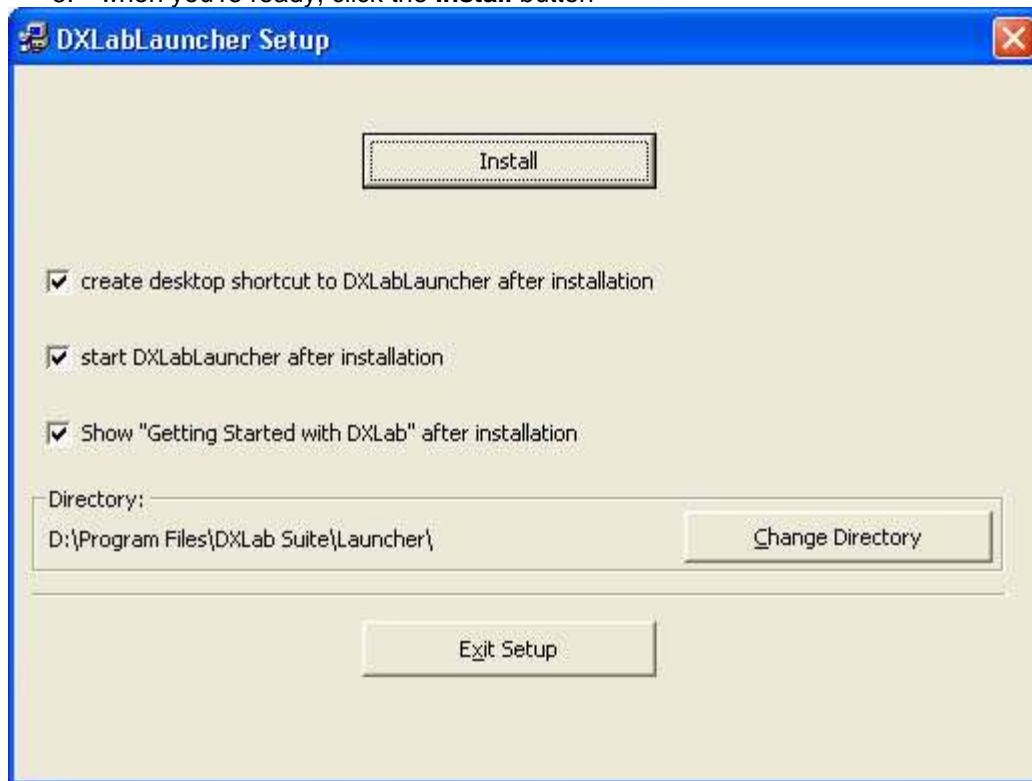


Several windows will flash by while the DXLabLauncher initializes. You should have already terminated any other running applications, so click the **OK** button when this Welcome window appears:

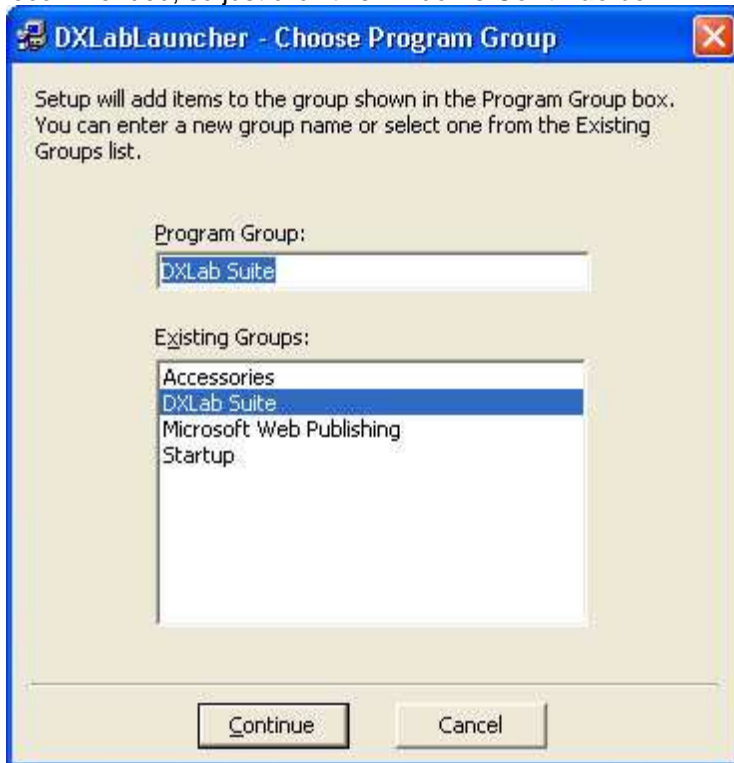


Before clicking the **Install** button in the window below,

1. leave the **create desktop shortcut...** box checked unless you **don't** want a desktop shortcut to the Launcher created; a desktop shortcut makes it easy to start the Launcher, which in turn can start any DXLab application, so most users will leave this box checked
2. leave the **start DXLabLauncher...** box checked
3. since you're already viewing **Getting Started with DXLab**, uncheck the **Show Getting Started...** box
4. if you want to install the Launcher in a folder **other than** the one shown in the **Directory** panel, then click the **Change Directory** button and select the desired destination folder
5. when you're ready, click the **Install** button



Placing start menu entries for DXLab applications in a group named **DXLab Suite** is recommended, so just click this window's **Continue** box:



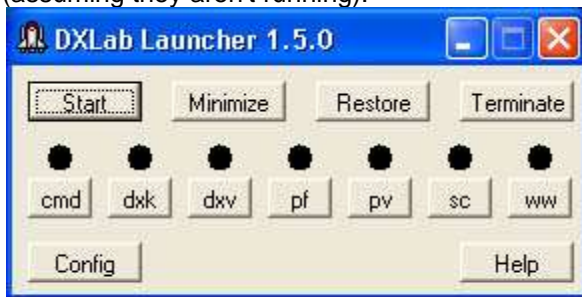
DXLabLauncher Setup will now install the Launcher, displaying its progress as it proceeds:



When the DXLabLauncherSetup completes, click the **OK** button:



The DXLab Launcher will now start. If no other DXLab applications are installed, the LED-like displays above each of the Launcher's application control buttons are shown in **black**. If you previously installed other DXLab applications, their LED-like displays will be shown in **red** (assuming they aren't running).



No DXLab applications installed



Pathfinder and WinWarbler already installed

To install an uninstalled application, click the appropriate Launcher button:

- **cmd** - Commander (transceiver control)
- **dxk** - DXKeeper (logging)
- **dxv** - DXView (DX information and antenna rotator control)
- **pf** - Pathfinder (QSL route discovery)
- **pv** - PropView (Propagation Prediction)
- **sc** - SpotCollector (DX and WWV spot collection and analysis)
- **ww** - WinWarbler (PSK31, PSK63, RTTY, CW, and Phone operation)

See <http://www.dxlabsuite.com/dxlabwiki/InstallApplication> for additional information.

To upgrade an installed DXLab application, see <http://www.dxlabsuite.com/dxlabwiki/UpgradeApplication>

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@yahogroups.com.

Configuration

Using the Launcher's **Configuration** window, you can

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 non-DXLab applications to be started before DXLab applications and terminated after DXLab applications
4. Specify non-DXLab applications to be started after DXLab applications and terminated before DXLab applications
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. Starting an individual non-DXLab application - as described here
8. Terminating an individual non-DXLab application - as described here
9. Indicate whether or not you want the Launcher's main window to always be "on top"
10. Indicate whether the Launcher's Main window should automatically minimize after the specified applications are started
11. Indicate whether the Workspaces window should be displayed on startup
12. Indicate whether the Launcher should check for new versions of installed applications on startup and daily thereafter
13. Direct the Launcher to immediately check for new versions of installed applications
14. Specify the web site or local folder from which applications will be installed or upgraded
15. Indicate whether the Launcher should automatically shut down after you click the Terminate button

To display the Launcher's **Configuration** window, click the Config button in the Launcher's Main window. The **Configuration** window contains three tabs: the **Apps Started Before DXLab Apps** tab, the **DXLab Apps** tab, and the **Apps Started After DXLab Apps** tab.

Specifying a DXLab application's pathname

To specify a DXLab application's pathname, locate the application's row on the **DXLab Apps** tab, 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
```

The pathname of the file you select will appear in the application's **Program Path** textbox.

If on startup, the Launcher finds an installed DXLab application whose **Program Path** textbox is empty, it will initialize this textbox with the pathname of the installed executable. If you have downloaded a later version of this application's executable to a different pathname, use the **Sel** button to select that pathname.

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 Apps** tab'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 non-DXLab applications to be started before DXLab applications and terminated after DXLab applications

The Launcher enables you to specify up to 8 non-DXLab applications that can be started in a specified order before you start a group of DXLab applications, and terminated after that group of DXLab applications are terminated. Initially, these are depicted on the **Apps Started Before DXLab Apps** tab as *Before App 1* through *Before App 8*. To specify the pathname of such an application, click the **Sel** button associated with an unused **Command Line** 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*.

You can optionally append command line switches to the Command line, e.g.

```
D:\HamApps\JTAlert\JTAlert.exe /wsjtx
```

Check the **Enab** box associated with any non-DXLab application that should be started when you click the Start button on the Launcher's Main window and terminated when you click the Terminate button on the Launcher's Main window.

- Check the **Start Minimized** box associated with any application that should be started with its Main window minimized; note that not all applications obey this directive.
- To delay the start of subsequent non-DXLab or DXLab applications for a specified interval after a non-DXLab application is launched, specify the duration of that interval in seconds in the **Delay** box associated with this application.
- Set the Termination selector to specify how the application should be terminated, either by clicking the Main window's Terminate button, or by clicking the application's individual Terminate button.

Specifying non-DXLab applications to be started after DXLab applications and terminated before DXLab applications

The Launcher allows you to specify up to 8 non-DXLab applications that can be started in a specified order after you start a group of DXLab applications. Initially, these are depicted on the **Apps Started After Other Applications** tab as *After App 1* through *After App 8*. To specify the *program pathname* of such an application, click the **Sel** button associated with an unused **Command Line** 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*.

You can optionally append *command line arguments* to the **Command Line's program path**, e.g.

```
D:\HamApps\JTAlert\JTAlert.exe /wsjtx
```

Check the **Enab** box associated with any non-DXLab application that should be started when you click the Start button on the Launcher's Main window and terminated when you click the Terminate button on the Launcher's Main window.

- Check the **Start Minimized** box associated with any application that should be started with its Main window minimized; note that not all applications obey this directive.
- Set the Termination selector to specify how the application should be terminated, either by clicking the Main window's Terminate button, or by clicking the application's individual Terminate button.

I
ndicating 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 **Enab** box in the **DXLab Apps** tab, **Apps Started Before DXLab Apps** tab, and **Apps Started After DXLab Apps** tab. If the **CTRL** key is depressed when the **Terminate** button is clicked, applications are terminated whether or not their associated **Enab** box is checked.

Automatically starting the specified applications

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

Starting an individual non-DXLab application

To start a non-DXLab application, locate its entry on the **Apps Started Before DXLab Apps** or **Apps Started After DXLab Apps** tab, and click its **Start** button.

Terminating an individual non-DXLab application

To terminate a non-DXLab application, locate its entry on the **Apps Started Before DXLab Apps** or **Apps Started After DXLab Apps** tab, and click its **Terminate** button; the application will be terminated using the method specified by its Termination selector. To terminate a non-DXLab application without being prompted for confirmation, depress the CTRL key while clicking its **Terminate** button.

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.

Indicating whether the Workspaces window should be displayed on startup

If you want the Launcher to display the Workspaces window on startup, then check the Configuration window's **Workspaces On Startup** box.

Automatically check for new versions of installed applications on startup and daily thereafter

If you want the Launcher to report the availability of new versions of installed DXLab applications on startup and daily thereafter, then check the **Check for New each day** box.

Immediately check for new versions of databases and installed applications

To immediately check for new versions of databases and installed DXLab applications, click the **New ?** button. To see what versions of the DXCC, IOTA, USAP, LOTW, and eQSL AG databases are presently installed and available, click the **Databases** button to display the Launcher's **Databases** window. Double-clicking the **DXCC Database's Installed** or **Available** version number will display its Change Notes.

The Launcher determines what versions of DXLab applications and databases are available by downloading a file from www.dxlabsuite.com. If the Launcher cannot access this site, it displays the message "Can't access internet" on its Main window; potential causes include

- problems at your Internet Service Provider
- www.dxlabsuite.com is inaccessible or not responding
- an anti-malware or firewall application is preventing the Launcher from accessing the internet
- internet access is disabled or misconfigured in Windows
- a cable is unplugged, or a hardware device is powered down or malfunctioning

Specifying the web site or local folder from which DXLab applications are installed or upgraded

Several web sites host DXLab development versions. Specify the site from which the Launcher will download new version via the **Distribution Site** selector. You can direct your web browser to navigate to the select **Distribution Site's** download web page by clicking the **Visit** button.

Alternatively, you can set the **Distribution Site** selector to *local folder* and set the **Install folder pathname** to designate a folder containing

- a file named `FullInfo.txt` that enumerates the available full (archive) releases formatted as shown in this example
- a file named `ReleaseInfo.txt` that enumerates the available update releases formatted as shown in this example
- a folder for each DXLab application, each of which contains the application's archive and update self-extracting executables; these folders must be named
 - Commander
 - DXKeeper
 - DXView
 - Pathfinder
 - PropView
 - SpotCollector
 - WinWarbler
 - DXLabLauncher

Automatically shutting down the Launcher after terminating the specified applications

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

Automatically terminate all specified applications when the Launcher is shut down

If you want all specified applications to be terminated when the Launcher is shut down, then check the **Configuration window's Terminate on Shutdown** 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.

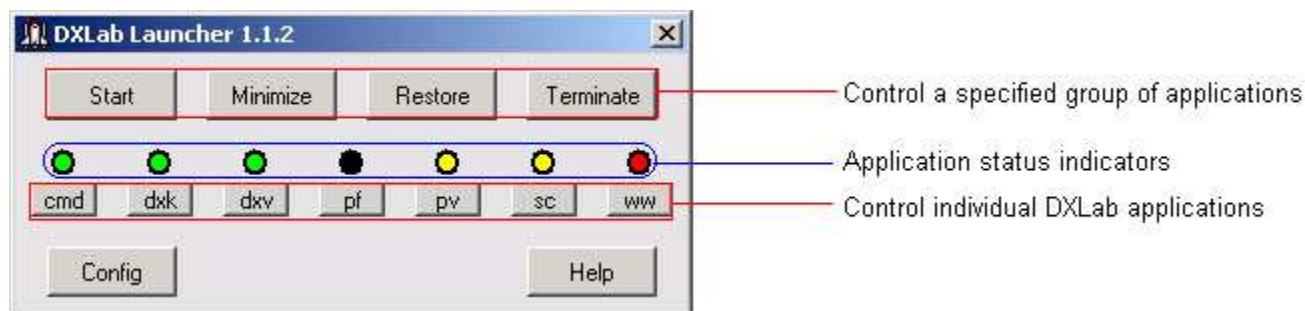
Other Settings

- **Use Multiple Monitors:** if not checked, windows that were displayed on a secondary monitor during the previous session will be restored to the primary 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.

Other Functions

- **Error Log** - when clicked, displays the `Errorlog.txt` file from the DXView folder
- **License** - when clicked, displays DXView's License
- **Help** - when clicked, displays DXView's reference documentation

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** that 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 -- DXLab and non-DXLab -- simultaneously

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

The Configuration window provides **Start** and **Terminate** buttons that enable you to start and terminate individual non-DXLab applications.

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.

If there are no application updates or database updates known to be available, the Launcher's Main window displays the name of the last Workspace activated in the space between the Config and Help buttons.

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
black	not installed
red	not running
cyan	starting
green	running, not minimized
yellow	running, minimized
white	terminating

Individual Application Control Buttons

Each of these buttons on the Launcher's Main window 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
black	not installed	user clicks Individual Control Button	not running	red
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** offers to install a DXLab application that isn't yet installed, and starts an installed 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. Depressing the **Ctrl** key while 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.

When a DXLab application starts, its windows are restored to their state (minimized or not) when the application was last terminated.

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**.

Clicking an application's **Status Indicator** is equivalent to clicking its **Individual Control Button**. You can optionally specify the pathname of a log file to be opened by DXKeeper on startup.

Group Control Buttons

Button	Function
Start	<ul style="list-style-type: none">start all applications - DXLab and non-DXLab - whose Enab box is checked, and presents the Abort button until all applications have been directed to start
Abort	<ul style="list-style-type: none">aborts the process of starting applications, and presents the Start button
Minimize	<ul style="list-style-type: none">hide all windows of all running DXLab applications
Restore	<ul style="list-style-type: none">restore the Main windows of all running DXLab applications
Terminate	<ul style="list-style-type: none">if the CTRL key is not depressed while clicking this button, terminates all applications - DXLab and non-DXLab -whose Enab box is checked; each non-DXLab application is terminated using the method specified by its Termination selectorif the CTRL key is depressed while clicking this button, terminate all applications - DXLab and non-DXLab - whether or not their Enab boxes are checked ; each non-DXLab application is terminated using the method specified by its Termination selector

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.

Order of Application Start-up and Termination

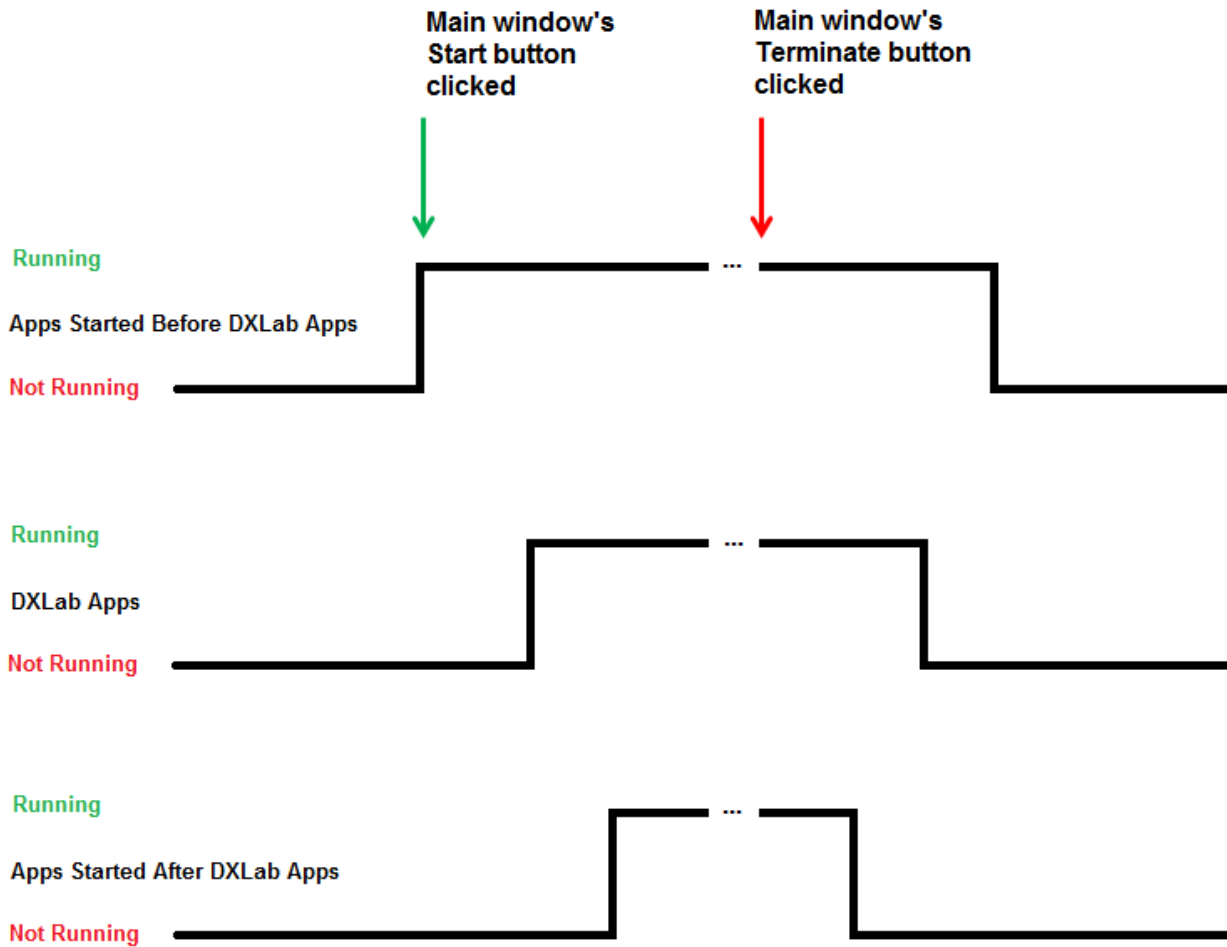
When the Main window's **Start** button is clicked, enabled applications are started in this order:

1. Apps Started Before DXLab Apps, in sequential order
2. DXLab Apps, in sequential order
3. Apps Started After DXLab Apps, in sequential order

When the Main window's **Terminate** button is clicked, enabled applications are terminated in this order:

1. Apps Started After DXLab Apps
2. DXLab Apps
3. Apps Started Before DXLab Apps

The timing of start-up and termination is illustrated in this timing diagram:



Starting non-DXLab applications

The Configuration window's **Apps Started Before DXLab Apps** and **Apps Started After DXLab Apps** tabs provide **Start** buttons that enable you to start individual non-DXLab applications by issuing a **Command Line**, whether or not their **Enab** box are checked.

- If the tab is displaying **Main Window Captions** for each application, click the **Show Command Lines** button to display **Command Line** text boxes so you can specify a **Command Line** for each application
- A **Command Line** that is composed of a *program pathname* specifying a file whose filename extension is .exe, .bat, .lnk, or .vxe may optionally be followed by one or more *command line arguments*; for example:
`C:\HamApps\JTAlert\JTAlert.exe /wsjtx`
- A **Command Line** that is composed of a *program pathname* specifying a file whose filename extension is other than .exe, .bat, .lnk, or .vxe may may not specify *command line arguments*

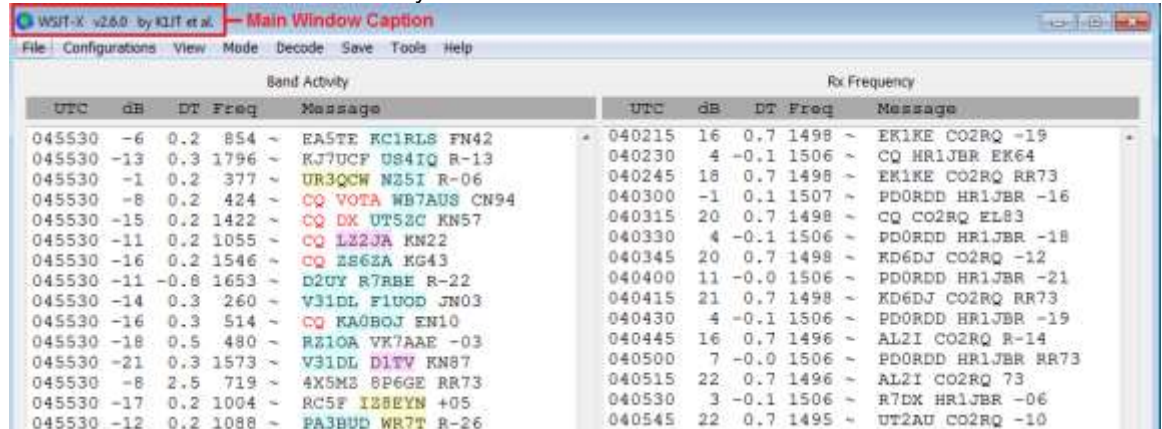
Terminating non-DXLab applications

The Configuration window's **Apps Started Before DXLab Apps** and **Apps Started After DXLab Apps** tabs provide **Terminate** buttons that enable you to terminate individual non-DXLab applications whose program pathname does not specify a file whose extension is .bat. Each application's **Termination** selector specifies the method by which the application should be terminated.

Most applications can be terminated using the **Kill Single** method; try that first. Do not use the **ForceKill** termination method with any application that manages data in databases or files.

- **None:** the application is not terminated when its **Terminate** button is clicked
- **Kill Single:** the Windows Taskkill command is invoked to send a *Close* message to the application's Main window, which for a Windows-compliant application should ensure that open files and databases are closed before the application terminates
 - the process name is extracted from the Command Line's program pathname
 - neither the */f* nor the */t* parameters are included in the Taskkill invocation
- **Kill:** the Windows Taskkill command is invoked to send a *Close* message to the application's Main window and to terminate any child processes created by the application's process
 - the process name is extracted from the Command Line's program pathname
 - the */t* parameter is included in the Taskkill invocation
- **ForceKill:** the Windows Taskkill command is invoked to *forcefully* terminate the application's process and to *forcefully* terminate any child processes created by the application's process
 - the process name is extracted from the Command Line's program pathname
 - the */f* and */t* parameters are included in the Taskkill invocation
 - when terminating an application using this method, its open files and databases may not be updated and closed before termination; note that not all applications open and update files or databases.
- **Close:** Windows is directed to send a *Close* message to the application's Main window, which for a Windows-compliant application should ensure that open files and databases are closed before the application terminates
 - Using this method requires exactly specifying the application's **Main Window Caption** as it appears in the title bar of the application's Main window, including upper/lower case characters, spaces, and punctuation.

To terminate WSJT-X using the Close method, for example, note that the WSJT-X Main window's caption has three spaces between the words "WSJT-X" and "v2.6.0" and three spaces between the words "v2.6.0" and "by":



This must be exactly specified in the **Main Window Caption** for WSJT-X, as shown in the screen shot below:



- If the tab is displaying a **Command Line** for each application, click the **Show Main Window Captions** button to display **Main Window Caption** text boxes so you can specify a **Main Window Caption** for each application
- Note: Using this method, the Launcher cannot terminate non-DXLab applications whose Main Window title bar content changes over time.
- **Kill Single:** the Windows Taskkill command is invoked to send a *Close* message to the application's Main window, which for a Windows-compliant application, should ensure that open files and databases are closed before the application terminates
 - the process name is extracted from the Command Line's program pathname
 - neither the */f* nor the */t* parameters are included in the Taskkill invocation
- **Kill:** the Windows Taskkill command is invoked to terminate the application's process and to terminate any child processes created by the application's process
 - the process name is extracted from the Command Line's program pathname
 - the */t* parameter is included in the Taskkill invocation
- **ForceKill:** the Windows Taskkill command is invoked to *forcefully* terminate the application's process and to *forcefully* terminate any child processes created by the application's process
 - the process name is extracted from the Command Line's program pathname
 - the */f* and */t* parameters are included in the Taskkill invocation
 - when terminating an application using this method, its open files and databases may not be updated and closed before termination; note that not all applications open and update files or databases.

To terminate a non-DXLab application without being prompted for confirmation, depress the CTRL key while clicking its **Terminate** button.

Each non-DXLab application's **Termination** selector also governs how it will be terminated if it's Enab box is checked when the Terminate button on the Launcher's Main window is clicked.

Displaying the Launcher's Configuration window

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

Installing, Upgrading, and Restoring DXLab Applications

Determining DXLab application installation status and version availability

To determine which DXLab applications are installed, and what versions are currently available, click the **Configuration window's New ?** button. If you want the Launcher to automatically update installation status and determine what versions are available versions when its first started and daily thereafter, then check the **Check for New each day** box.

DXLab applications that are not installed are shown with black **application status indicators** on the Main window, and in red font in the **Application** column of the **Configuration window's DXLab Apps** tab.

If you uninstall a DXLab application, click the **New ?** button to update the Launcher's display of installed applications.

The **Configuration window's DXLab Apps** tab provides one row for each DXLab application. The **Action** column in this row provides buttons that enable you to install, upgrade, or restore a DXLab application:

- Install - clicking this button will install an application that has not yet been installed
- Upgrade - clicking this button will upgrade an application that has been installed, but for which a more recent version is available
- Restore - clicking this button will restore an application to the current version; this is of use if you have inadvertently deleted one or more of the application's files, or if you have been running a not-yet-released version and wish to return to the current version

Installing DXLab applications

If a DXLab application has not been installed on your PC, its name will be displayed in red font in the **Configuration window's DXLab Applications** panel. If you have selected a **Distribution Site** or a **local folder**, then clicking the application's **Install** button in the **DXLab Apps** tab will

1. let you choose a folder into which the application should be installed, defaulting to the folder specified in the application's **Program Path** in the **Configuration window's DXLab Apps** tab
2. download the full release's zip archive from the selected **Distribution Site** or copy the full release's zip archive from the specified **local folder**
3. extract all files in the zip archive and run the application's setup program to properly deploy them in the selected installation folder; this setup program will appear on screen and will display instructions that you must follow
4. download, extract, and deploy any additional files required by the application
5. ask you whether to download, extract, and deploy any optional files used by the application
6. update the application's **Program Path**
7. determine whether an upgrade is available and if so apply it

You can also initiate the installation of an uninstalled DXLab application by clicking its **Individual Application Control Button** on the Main window.

Upgrading installed DXLab applications

The version numbers of both **Installed** and **Available** releases are shown on **Configuration window's DXLab Apps** tab; double-clicking a version number will display its **Release Notes**. If a new release of an installed DXLab application is available and you have selected a **Distribution Site** or a **local folder**, then clicking the application's **Upgrade** button in the **Configuration window's DXLab Apps** tab will

1. inform you if the application in question is currently running so you can terminate the application before re-initiating the upgrade operation
2. download the newly available version's zip archive from the selected **Distribution Site** or copy the newly available version's zip archive from the specified **local folder**
3. extract all files in the zip archive and properly deploy them in the folder specified by the **Program Path** in the **Configuration window's DXLab Apps** tab
4. create a copy of the application's executable without an embedded version number (e.g. `DXKeeper.exe` or `WinWarbler.exe`) and if necessary update the **Program Path** in the **Configuration window's DXLab Apps** tab to refer to this executable
5. delete the zip archive downloaded in step 2

Restoring installed DXLab applications

If you are running a version of an installed DXLab application that is the same as or more advanced than the currently available version and you have selected a **Distribution Site**, then clicking the application's **Restore** button in the **Configuration window's DXLab Apps** tab will

1. inform you if the application in question is currently running so you can terminate the application before re-initiating the restore operation
2. download the currently available version's zip archive from the selected **Distribution Site** or copy the currently available version's zip archive from the specified **local folder**
3. extract all files in the zip archive and properly deploy them in the folder specified by the **Program Path** in the **Configuration window's DXLab Apps** tab
4. create a copy of the application's executable without an embedded version number (e.g. `DXKeeper.exe` or `WinWarbler.exe`) and if necessary update the **Program Path** in the **Configuration window's DXLab Apps** tab to refer to this executable
5. delete the zip archive downloaded in step 2

Uninstalling DXLab applications

The Launcher cannot uninstall a DXLab application; to do this, open the **Windows Control Panel**, run the **Add or Remove Programs** applet, select the DXLab application you wish to uninstall, and then click the **Remove** button. Then click the Launcher's **New ?** button to update its display of installed DXLab applications.

Using Workspaces to Save and Load Windows Registry Settings

In conformance with Windows programming standards, DXLab applications store their settings in the Windows Registry - with one exception: DXKeeper stores some settings in the current Log file. Windows Registry 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 Windows Registry settings to a file, and to later load the Windows Registry with settings from that file:

1. for archival storage, to be restored after a 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 Loading DXLab application settings with Workspaces

The Launcher enables you to save the Windows Registry settings of all specified DXLab applications into a *Workspace* - a folder in a specified location. Each application's settings are stored within the workspace folder in an appropriately named file. Clicking the Configuration window's **Workspaces** button or depressing the **CTRL** key while clicking the Main window's **Config** button displays the Launcher's **Workspaces** window; if **WS: WorkspaceName** is displayed on the Main window, clicking it will also display the Workspace window.

The **Workspaces (in Current Workspace Folder)** panel displays a list of the Workspaces that are present in the folder whose pathname is specified in the **Current Workspace Folder** box; by default, this is set to the Launcher's *Workspaces* folder. You can choose a different folder by clicking the **Select Folder** button to the right of the Workspaces window's **Current Workspace Folder** box. Letting the mouse cursor hover over a Workspace in the **Workspaces (in Current Workspace Folder)** panel will display a popup window showing the name of each DXLab application for which settings are stored in that Workspace.

To select a Workspace, click its entry in the **Workspaces (in Current Workspace Folder)** panel.

The Launcher's Workspaces window provides four primary functions: **Create Empty Workspace**, **Save Settings to Workspace**, **Load Settings from Workspace**, and **Delete Workspace**; the latter three functions are disabled if no Workspace is selected in the **Workspaces (in Current Workspace Folder)** panel:

Create Empty Workspace	<p>prompts you to specify a Workspace name, and then creates the Workspace</p> <ul style="list-style-type: none">• since Workspaces are folders, Workspace names must conform to the rules for folder names• after creation, the Workspace contains no Windows Registry settings
Save Settings to Workspace	<p>saves the Windows Registry settings of all specified DXLab applications into files in the selected Workspace, over-writing any existing files of the same name in that Workspace</p> <ul style="list-style-type: none">• terminate all running DXLab applications except the Launcher before invoking this function• for each DXLab application with a valid Program Path in the Configuration window's DXLab Applications panel, Windows Registry settings are saved to a file in the Workspace
Load Settings from Workspace	<p>loads the Windows Registry with settings from each file contained in the selected Workspace</p> <ul style="list-style-type: none">• terminate all running DXLab applications except the Launcher before invoking this function• the Launcher will immediately load its settings from the selected Workspace <p>The name of the Workspace from which Windows Registry settings were most recently loaded is shown between square brackets in the Configuration window's title bar; if no application or database updates are available, the name of the Workspace from which Windows Registry settings were most recently loaded is also displayed on the Launcher's Main window as WS: WorkspaceName .</p>
Delete Workspace	<p>deletes the selected Workspace</p>

Backing Up Windows Registry Settings for All DXLab applications to a Workspace

To facilitate recovery should your PC subsequently suffer hardware or software problems,

1. specify pathnames for all of your DXLab applications on the Configuration window's DXLab Applications panel
2. terminate all running DXLab applications except the Launcher
3. in the Launcher's Workspaces window, click the **Create Empty Workspace** button and specify the name of the Workspace to be created
4. in the Workspaces window's Workspaces panel, click the name of the newly-created Workspace to select it
5. in the Workspaces window, click the **All DXLab Applications** panel's **Save Settings to Workspace** button
6. configure your Backup and Recovery application to include the Launcher's Workspaces folder (and its sub-folders)

Recovering Windows Registry Settings for All DXLab applications from a Workspace

After repairing the root cause of a hardware or software failure that caused the loss of settings stored in the Windows Registry,

1. direct your Backup and Recovery application to restore the Launcher's Workspaces folder (and its sub-folders) and any other folders containing DXLab data (e.g. your log file)
2. install the Launcher and the other DXLab applications you use
3. start the Launcher; do not start any other DXLab applications
4. in the Workspaces window's Workspaces panel, select the Workspace whose settings are to be loaded into the Windows Registry by clicking its name
5. in the Workspaces window, click the **All DXLab Applications** panel's **Load Settings from Workspace** button

Backing Up Windows Registry Settings for One DXLab application to a Workspace

If you've intentionally changed one DXLab application's settings, you can back up that application's settings to a Workspace, replacing any previously backed-up settings in that Workspace for that application:

1. terminate the DXLab application whose settings you wish to back up
2. in the Workspaces window
 - o in the **Workspaces** panel, click on the name of the Workspace that will receive the backed up settings
 - o in the **Selected DXLab Application** panel,
 - set the **Application Settings to Save** selector to the name of the DXLab application whose settings will be backed up
 - click the **Save Settings to Workspace** button

Recovering Windows Registry Settings for One DXLab application from a Workspace

After inadvertently modifying the settings of one DXLab application, you can restore that application's settings to the Windows Registry from a Workspace:

1. terminate the DXLab application whose settings you wish to recover
2. in the Workspaces window
 - o in the **Workspaces** panel, click on the name of the Workspace from which settings will be recovered
 - o in the **Selected DXLab Application** panel,
 - set the **Application Settings to Load** selector to the name of the DXLab application whose settings will be recovered
 - o click the **Load Settings from Workspace** button

Specifying a Workspace from which Windows Registry Settings are to be Loaded When the Launcher is Started via a Desktop Icon or Batch File

If you start the Launcher via a batch file or a desktop icon, appending the name of a Workspace to the command line will cause the Launcher to load Windows Registry settings from that Workspace on startup. If the name of the workspace includes spaces, enclose the Workspace name with double-quote characters, e.g. "contest settings".

Using Workspaces to Switch Between Multiple Operating Configurations

To establish multiple operating configurations,

1. setup your DXLab applications for the first configuration
2. terminate all DXLab applications except the Launcher; the Launcher's **Terminate** button is a convenient way to do this
3. in the Launcher's Workspaces window, click the **Create Empty Workspace** button and specify the name of the Workspace to be created; choose a name associated with the configuration whose settings it will save
4. in the Workspaces window's Workspaces panel, click the name of the newly-created Workspace to select it
5. in the Workspaces window, click the **All DXLab Applications** panel's **Save Settings to Workspace** button
6. 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. in the Workspaces window's Workspaces panel, select the Workspace for those configuration whose settings are to be loaded into the Windows Registry by clicking its name
3. in the Workspaces window, click the **All DXLab Applications** panel's **Load Settings from Workspace** button

Saving, Clearing, and Restoring Windows Registry settings for Individual DXLab applications

The **Registry** panel on the **DXLab Apps** tab of the 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. Clicking an application's **Sv** button saves that application's registry settings to a file; clicking an application's **Ld** button loads that application's settings from a file. When invoking an application's registry save or registry load functions, the application should not be running. The Launcher will immediately update the applications settings from the registry after completing the **Ld** operation.

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 clicked the Launcher's **Ld** button. 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** 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.