

Plugins

You can extend the capabilities of Source by adding new component models or data processing tools. Some of these "plugin" tools extend Source's user interface. Other plugins, especially those written by third parties, may look and operate differently from the normal user interface. Refer to [Assign catchment models](#) when working with catchment model plugins.

Plugins are files which have a .DLL or .EXE extension, and work with specific versions of Source. Plugins that are external to Source, and must be downloaded from the Internet, must be handled differently. Refer to [Externally created plugins](#) for more information..

Note: Many of the existing eWater CRC Toolkit tools (or components of these tools) can be used as Source plugins; however, you must ensure that the plugin version is compatible with the Source version. In other words, the plugin .dll files have been compiled against the Source version that the plugin is to be used with. Also note that plugins are compatible with different version of Source, provided that the first three digits of the Source version number are the same. For example, a plugin that is compatible with different versions of Source 3.7.0 will not be compatible with 3.5.0.

There are several plugins included with Source that have been verified by the Source development team. This documentation is only developed and maintained for these included plugins. Table 1 provides plugin file and location details of plugins that are discussed in more detail.

Table 1. Plugins

Plugin	Plugin file	Location in Source
GWLag Plugin	C:\Program Files\ewater\<Source version>\Plugins\RiverSystem.Plugins.PerfectGWLag.dll	Tools » Plugins » <various related folders>
River Analysis Package	C:\Program Files\ewater\<Source version>\Plugins\RiverSystem.Plugins.RAP.dll	Tools » Plugins » Ecology.RAP » <name of RAP tool>
Spatial data pre-processor	C:\Program Files\ewater\<Source version>\Plugins\RiverSystem.Plugins.SpatialDataPreProcessor.dll	Tools » Plugins » RiverSystem.Plugins.SpatialDataPreProcessor

Note: Follow these guidelines when working with plugins in Source:

- No projects must be open when loading plugins; and
- Restart Source when unloading plugins (ensure that they are enabled in the Plugin Manager first).

Working with the Plugin Manager

The Plugin Manager allows you to install, load and manage all plugins in Source. Available via **Tools » Plugin Manager**, this dialog (shown in Figure 1) shows all plugins that have been installed either:

- Manually; or
- Via the online store hosted by eWater.

The Details panel in the Plugin Manager provides the following information for the selected plugin:

- Version - a four-digit number used to uniquely identify which version of Source the plugin was compiled against; and
- Tags - the version of Source that the plugin can be opened in.

Installing plugins

Manually installing plugins

Source community plugins are installed as part of Source are located in the following folder:

C:\Program Files\eWater\

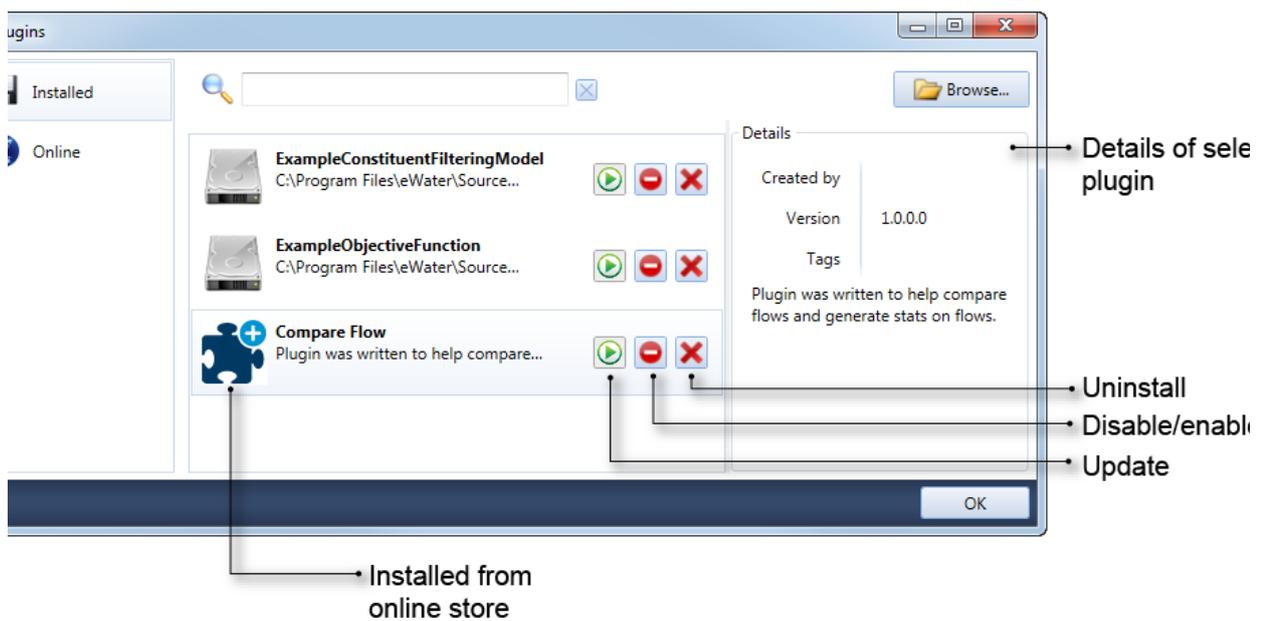
where:

<Source version> refers to the version of Source currently in use.

To install a plugin manually:

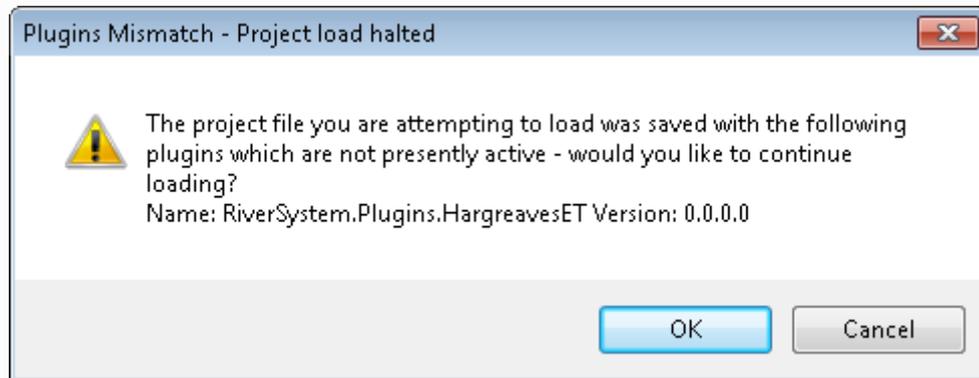
- Click **Browse...** to open an Explorer dialog;
- Select the plugin; and
- Click **OK**.

Figure 1. Plugin manager



Note: If a required plugin is not loaded, Source might have a problem running. In this instance, you will be notified with a warning dialog (example in Figure 2).

Figure 2. Error without installed plugin



Installing via the online store

- Choose **Online** to view a complete list of plugins available from the store; and
- Click **Install**.

Uninstalling plugins from a source project

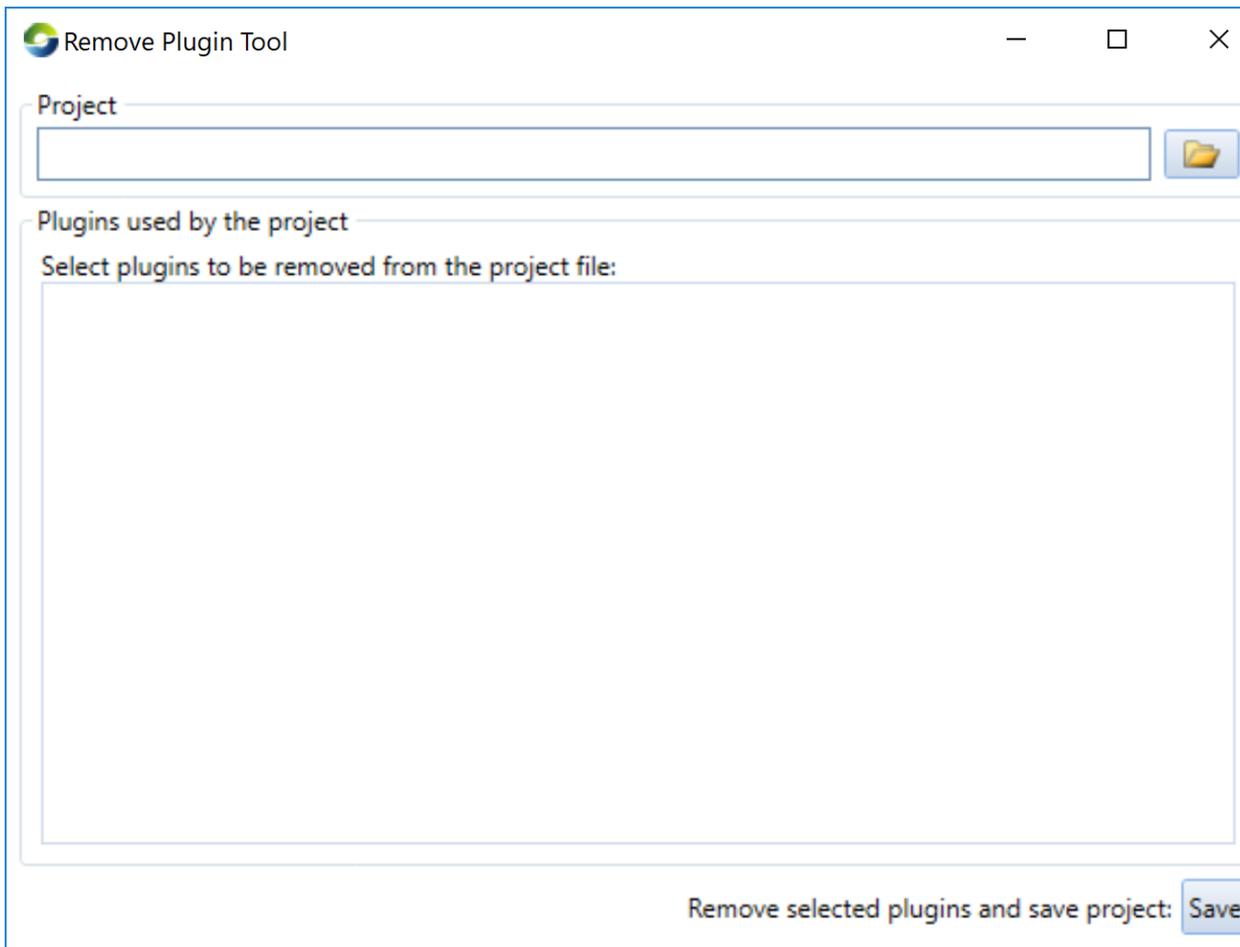
To uninstall plugins from a source project when they have been added accidentally, normally just load this project file to a source version which does not install the plugins (not included in plugin manager) and then save the project. By doing this, source will try to remove the plugins. But for certain plugins, the following steps are needed to be able to uninstall it from a source project:

- Go to **Tools>>Remove Plugin from project file.**
- Add the project and then select the plugin from the lower window under "Select plugins to be removed from the project file", see figure 3.
- Click "**Save**" to save the project.
- If you haven't already removed the plugin from source plugin manager, go to **Tools >> Plugin Manager >> uninstall "the plugin"** , click OK and restart Source when prompted. Now you should be able to load the project file to source without the plugin loaded.

This is currently only required for plugins:

- NetLP Monthly Optimiser

Figure 3. Remove Plugin Tool



Upgrading plugins

There are two possible scenarios that warrant an upgrade:

- The version of Source has been upgraded - in this case, Source will attempt to install the same plugin version that was installed in the previous version of Source; or
- The plugin version is different - if the plugin was originally installed from the online store, clicking **Update** will remove the old version and install the new one. Note that this happens only once when Source is opened for the first time with the new plugin version.

Writing your own plugins

To write your own plugins using .NET and C#, contact support@ewater.org.au. With a Source licence, you can get access to [How to write plugins for Source](#).

Externally created plugins

For plugins that have been developed externally to Source (such as DERMTTools), you must manually unblock them in Windows after downloading. By default, Windows blocks the execution of programs downloaded from the internet (including the .dlls used as Source plugins) in order to reduce the risk of the computer being infected by a virus or some other form of malicious software.

 The use of the unblock option carries with it risk, in that it removes a key defence against malicious software. You should only use unblock a DLL or EXE when you are confident that the software in question does not contain malicious code.

Once downloaded, open as follows:

- Open Explorer. Right click on the dll file and choose **Properties**; and
- In the resulting dialog (shown in Figure 4), click **Unblock** at the bottom of the screen, then **OK**.

The plugin can then be loaded as normal in Source.

Figure 4. Unblocking external plugins

