

Invisible Procedure Files

If you create a package of Igor procedures to be used by regular Igor users (as opposed to programmers), you may want to hide the procedures to reduce clutter or to eliminate the possibility that they might inadvertently change them. You can do this by making the procedure files invisible.

Invisible procedure files are omitted from Igor's Procedure Windows submenu which appears in the Windows menu. This keeps them out of the way of regular users.

There are three ways to make a procedure file invisible. In order of difficulty they are:

- Using the `#pragma hide` compiler directive
- Using an independent module
- Using the operating-system-supplied file visibility property

Invisible Procedure Windows Using `#pragma hide`

You can make a procedure file invisible by inserting this compiler directive in the file:

```
#pragma hide=1
```

This prevents the procedure window from being listed in the Windows→Procedures submenu. Procedures windows that include this compiler directive become invisible on the next compile.

You can make these windows visible during development by executing:

```
SetIgorOption IndependentModuleDev=1
```

and return them to invisible by executing:

```
SetIgorOption IndependentModuleDev=0
```

You must force a compile for this to take effect.

Prior to Igor Pro 6.30 this feature worked for `#included` procedure files only, not for packed and standalone procedure files.

Invisible Procedure Windows Using Independent Modules

You can also make a set of procedure files invisible by making them an independent module. The independent module technique is more difficult to implement but has additional advantages. For details, see **The IndependentModule Pragma** on page IV-55.

Invisible Procedure Files Using The Files Visibility Property

This section discusses making procedure files invisible by setting the operating-system-supplied file "visible" property.

Note: This is an old technique that is no longer recommended. It may not be supported in future versions of Igor.

When Igor opens a procedure file, it asks the operating system if the file is invisible (*Macintosh*) or hidden (*Windows*). We will use the term "invisible" to mean invisible on Macintosh and hidden on Windows.

If the file is invisible, Igor makes the file inaccessible to the user. Igor checks the invisible property only when it opens the file. It does not pay attention to whether the property is changed while the file is open.

You create Igor procedures using normal visible procedure files, typically all in a folder or hierarchy of folders. When it comes time to ship to the end user, you set the files to be invisible. If you set a file to be invisible, you should also make it read-only.