The use of backslashes is complicated by the fact that Igor uses the backslash character as an escape character in literal strings. This is also described in detail under **Path Separators** on page III-451. The simplest solution to this problem is to use colon to separate path elements, even when you are running on Windows.

If you are writing procedures that need to extract sections of file paths or otherwise manipulate file paths, the **ParseFilePath** function may come in handy.

Symbolic Paths

A symbolic path is an Igor object that associates a short name with a folder on a disk drive. You can use this short name instead of a full path to specify a folder when you load, open or save a file. We recommend using symbolic paths instead of full paths when possible because they are more easily changed if your disk organization changes or if you want to access another location on disk.

Igor creates symbolic paths for the "Igor Pro Folder", the "Igor Pro User Files" folder and for the current experiment's home folder automatically. These symbolic paths exist in all experiments.

You can also create custom symbolic paths pointing to any folder on disk. These symbolic paths are part of the current experiment and cease to exist when you create a new experiment.

The New Symbolic Path Dialog

To access the New Symbolic Path dialog, choose New Path from the Misc menu. Use of this dialog is illustrated in the next section.

Symbolic Path Example

This example illustrate why you should use symbolic paths and how to use them. We assume that you have a folder full of text files containing data that you want to graph in Igor and that the organization of your hard disk is as follows:

```
hd or C:
   Users
      Jack
         Documents
            Data
               2016
                  May
                  June
                  July
```

Assume that we want to access files in the June folder.

To create a symbolic path for the folder:

1. Choose New Path from the Misc menu. This displays the New Symbolic Path dialog.
2. Enter Data in the Name field.
3. Click the Path button and locate the June folder.
4. Check the Overwrite checkbox.
5. Click Do It to create the symbolic path.

The NewPath command created by the dialog makes a symbolic path named Data which references:

```
hd:Users:Jack:Documents:Data:2016:June     (Macintosh)
C:\Users\Jack\Documents\Data\June           (Windows)
```

The command generated by the dialog uses Macintosh-style paths with colons separating components:

```
NewPath/O Data, "hd:Users:Jack:Documents:Data:2016:June"     // Macintosh

NewPath/O Data, "C:Users:Jack:Documents:Data:2016:June"      // Windows
```