

By clicking the Use Common Format radio button, you can choose from a pop-up menu of common formats. After choosing a common format, you can still control minor properties of the format, such as whether to use 2 or 4 digits years and whether to use leading zeros or not.

In the rare case that your file's date format does not match one of the common formats, you can use a full custom format by clicking the Use Custom Format radio button. It is best to first choose the common format that is closest to your format and then click the Use Custom Format button. Then you can make minor changes to arrive at your final format.

When you use either a common format or a full custom format, the format that you specify must match the date in your file exactly.

When loading data as delimited text, if you use a date format containing a comma, such as "October 11, 1999", you must make sure that LoadWave operation does not treat the comma as a delimiter. You can do this using the Delimited Text Tweaks dialog.

When loading a date format that consists entirely of digits, such as 991011, you should use the LoadWave/B flag to specify that the data is a date. Otherwise, LoadWave will treat it as a regular number. The /B flag can not be generated from the dialog — you need to use the LoadWave operation from the command line. Another approach is to use the dialog to generate a LoadWave command without the /B flag and then specify that the column is a date column in the Loading Delimited Text dialog that appears when the LoadWave operation executes.

Column Labels

Each column may optionally have a column label. When loading 1D waves, if you read wave names and if the file has column labels, Igor will use the column labels for wave names. Otherwise, Igor will automatically generate wave names of the form wave0, wave1 and so on.

Igor considers text in the label line to be a column label if that text can not be interpreted as a data value (number, date, time, or datetime) or if the text is quoted using single or double quotes.

When loading a 2D wave, Igor optionally uses the column labels to set the wave's column dimension labels. The wave name does not come from column labels but is automatically assigned by Igor. You can rename the wave after loading if you wish.

Igor expects column labels to appear in a row of the form:

```
<label><delimiter><label><delimiter>...<label><terminator>
```

where <column label> may be in one of the following forms:

<label>	(label with no quotes)
"<label>"	(label with double quotes)
'<label>'	(label with single quotes)

The default delimiter characters are tab and comma. There is a tweak (see **Delimited Text Tweaks** on page II-136) for using other delimiters.

Igor expects that the row of column labels, if any, will appear at the beginning of the file. There is a tweak (see **Delimited Text Tweaks** on page II-136) that you can use to specify if this is not the case.

Igor cleans up column labels found in the file, if necessary, so that they are legal wave names using standard name rules. The cleanup consists of converting illegal characters into underscores and truncating long names to the maximum of 255 bytes.

Examples of Delimited Text

Here are some examples of text that you might find in a delimited text file. These examples are tab-delimited.