

will not necessarily be of equal length. If you enable the “Ignore blanks at the end of a column” tweak then LoadWave will not load blanks at the end of a column into the 1D wave. If this option is enabled and a particular column has nothing but blanks then the corresponding wave is not loaded at all.

Troubleshooting Delimited Text Files

You can examine the waves created by the Load Delimited Text routine using a table. If you don’t get the results that you expected, you can try other LoadWave options or inspect and edit the text file until it is in a form that Igor can handle. Remember the following points:

- Igor expects the file to consist of numeric values, text values, dates, times or date/times separated by tabs or commas unless you set tweaks to the contrary.
- Igor expects a row of column labels, if any, to appear in the first line of the file unless you set tweaks to the contrary. It expects that the column labels are also delimited by tabs or commas unless you set tweaks to the contrary. Igor will not look for a line of column labels unless you enable the Read Wave Names option for 1D waves or the Read Column Labels options for 2D waves.
- Igor determines the number of columns in the file by inspecting the column label row or the first row of data if there is no column label row.

If merely inspecting the file does not identify the problem then you should try the following troubleshooting technique.

- Copy just the first few lines of the file into a test file.
- Load the test file and inspect the resulting waves in a table.
- Open the test file as a notebook.
- Edit the file to eliminate any irregularities, save it and load it again. Note that you can load a file as delimited text even if it is open as a notebook. Make sure that you have saved changes to the notebook before loading it.
- Inspect the loaded waves again.

This process usually sheds some light on what aspect of the file is irregular. Working on a small subset of your file makes it easier to quickly do some trial and error investigation.

If you are unable to get to the bottom of the problem, email a zipped copy of the file or of a representative subset of it to support@wavemetrics.com along with a description of the problem. Do not send the segment as plain text because email programs may strip out or replace unusual control characters in the file.

Loading Fixed Field Text Files

A fixed field text file consists of rows of values, organized into columns, that are a fixed number of bytes wide with a carriage return, linefeed, or carriage return/linefeed sequence at the end of the row. Space characters are used as padding to ensure that each column has the appropriate number of bytes. In some cases, a value will fill the entire column and there will be no spaces after it.

FORTRAN programs typically generate fixed field text files. A normal Fortran data file contains consists of values followed by spaces to pad to the field width. For example, the contents of a file using a field width of 10 might look like this (using dashes to represent spaces for clarity):

```
0.000-----1.000-----2.000-----<CRLF>
1.000-----2.000-----3.000-----<CRLF>
```

Non-Fortran programs sometimes write fixed-field data right-justified instead of left-justified, like this (using dashes to represent spaces for clarity):

```
-----0.000-----1.000-----2.000<CRLF>
-----1.000-----2.000-----3.000<CRLF>
```

To accommodate such files, Igor's Load Fixed Field routine strips leading and trailing spaces from the field before reading the value.