The field widths specified via W= override the default field width specified by the /F flag. If all of the columns in the file have the same field width then you can use just the /F flag.

You can load a subset of the columns in the file using the /L flag. Even if you do this, the column info specifications that you provide via the /B flag start from the first column in the file, not from the first column to be loaded.

Other LoadWave Issues

This section discusses other issues that apply to the LoadWave operation.

LoadWave Text Encoding Issues

This section discusses LoadWave text encoding issues of interest to advanced users. It assumes that you are familiar with the general topic of text encodings as explained under **Text Encodings** on page III-459.

Since Igor stores all text internally as UTF-8, it must convert text read from a file from the source text encoding to UTF-8. In order to do this it needs to know the source text encoding.

When loading an Igor binary wave file LoadWave ignores the /ENCG=*textEncoding* flag. The loaded wave's text encoding is determined as described under **LoadWave Text Encodings for Igor Binary Wave Files** on page III-475. The rest of this section applied to loading data from plain text files, not from Igor binary wave files.

When loading a text data file you can use the /ENCG=*textEncoding* flag to tell Igor what that text encoding is. See **Text Encoding Names and Codes** on page III-490 for a list of accepted values for textEncoding.

LoadWave uses the text encoding specified by /ENCG and the rules described under **Determining the Text Encoding for a Plain Text File** on page III-467 to determine the source text encoding for conversion of the text file's data to UTF-8. If you omit /ENCG or specify /ENCG=0, the specified text encoding is unknown and does not factor into the determination of the source text encoding. If following the rules does not identify a text encoding that works for converting the file's text to UTF-8, Igor displays the Choose Text Encoding dialog.

If the file contains nothing but ASCII characters, as is often the case, then any byte-oriented text encoding will work and there is no need to use the /ENCG flag.

When you are loading a huge file (e.g., hundreds of megabytes), finding a valid source text encoding may add a noticeable amount to the time it takes to load the file. If you know that the file is either all ASCII or is valid UTF-8, you can tell LoadWave to skip text encoding conversion altogether using an optional parameter, like this:

```
/ENCG={1,4}
```

"1" tells LoadWave that the text is valid as UTF-8, meaning that it is all ASCII or, if it contains non-ASCII characters, they are properly encoded as UTF-8.

"4" tells LoadWave to assume that the text is valid as UTF-8 and skip all validation and conversion.

In testing with a 200 MB delimited text file containing 1 million rows and 20 columns, we found that using /ENCG={1,4} saved about 10% of the time.

NOTE: If you use this flag but the file is not valid UTF-8 and you are loading data into text wave, the text waves will wind up with invalid data which will result in errors when you use the waves later.

As noted above, if following the rules does not identify a text encoding that works for converting the file's text to UTF-8, Igor displays the Choose Text encoding dialog. If you are loading many files using an unattended, automated procedure, displaying this dialog will cause your procedure to grind to a halt. You can prevent this by using another optional flag, like this:

```
/ENCG={1,8}
```