```
// /ONLY=0 means apply the command only to wave elements currently marked as
unknown.
SetWaveTextEncoding /DF={root:,1} /ONLY=0 3, 31
```

Now all elements of all waves are set to Windows-1252 except for the content element of text waves containing binary data which are so marked. In other words, there are no unknown text encodings and all elements are correctly marked. Marking all waves correctly ensures that Igor can correctly convert the wave plain text elements to UTF-8 for internal use.

If you were starting from a MacRoman experiment (Macintosh western text), you would use 2 instead of 3. If you were starting from a Japanese experiment (Shift JIS), you would use 4 instead of 3.

NOTE: The heuristic used by the /BINA flag is not foolproof. See **Text Waves Containing Binary Data** on page III-475 for details.

At this point you may want to convert your waves from Windows-1252 to UTF-8. This provides two advantages. First, Igor will not need to convert to UTF-8 when fetching the text waves' contents since the waves will already be UTF-8. Second, you will be able to use a wider repertoire of characters. The disadvantage is that you will get gibberish for non-ASCII characters if you open the experiment in any pre-Igor7 version. Assuming that you do want to convert to UTF-8, you would execute this:

```
// 1 means UTF-8. 31 means "all wave elements".
// /CONV means "convert text encoding". It automatically skips text wave
// content marked as binary.
SetWaveTextEncoding /DF={root:,1} /CONV=1 1, 31
```

Data Folder Name Text Encodings

Igor Pro 7.00 through 7.01 were unable to correctly load non-ASCII data folder names from Igor6 packed experiment files. Starting with version 7.02, in most cases, Igor correctly loads such data folder names.

If Igor mangles a non-ASCII data folder name when loading an Igor6 experiment, try choosing the correct text encoding from the Misc→Text Encoding→Default Text Encoding menu and turn Misc→Text Encoding→Default Text Encoding→Override Experiment on. Remember to restore these settings to their original settings after loading the experiment.

Starting with Igor Pro 7.02, Igor writes non-ASCII data folder names to packed experiment files using the text encoding of the built-in procedure window if possible. This will be Igor6-compatible if that text encoding is MacRoman on Macintosh or Windows-1252 on Windows. Igor Pro 7.00 through 7.01 expect UTF-8 and will therefore misinterpret such data folder names unless the text encoding of the built-in procedure window is UTF-8.

String Variable Text Encodings

Unlike the plain text elements of waves, there is no text encoding setting for string variables. Consequently Igor treats a string variable as if it contains UTF-8 text when printing it to the history area or when displaying it in an annotation or control panel or otherwise treating it as text.

If you have string variables in Igor6 experiments that contain non-ASCII characters, they will be misinterpreted by Igor7 or later. This may cause gibberish text or it may cause a Unicode conversion error. In either case you must manually fix the problem by storing valid UTF-8 text in the string variable.

For a local string variable, you can achieve this by assigning a new value to the string variable or by converting its contents from its original text encoding to UTF-8 using the **ConvertTextEncoding** function.

For a global string variable, you can achieve this by converting its contents from its original text encoding to UTF-8 using the **ConvertGlobalStringTextEncoding** operation. You need to know the original text encoding. It will typically be MacRoman if the Igor6 experiment was created on Macintosh, Windows-1252 if it was created on Windows, or Shift JIS if it was created using a Japanese version of Igor.