

Chapter II-5 — Waves

Igor displays the date/time wave in date format because it has "dat" units and all of the time values are 00:00:00. We now change the column format to date/time:

```
ModifyTable format(testDateTime)=8 // Set column to date/time format
```

Finally, we add the time:

```
testDateTime = testDateTime + testTime // Add time to date
```

To check the data type of your waves, use the info pane in the Data Browser. The data type shown for date/time waves should be "Double Float 64 bit". If not, use Data→Redimension Waves to redimension as double-precision.

So far we have looked at storing dates in the data of a wave. Typically such a date wave is used to supply the X wave of an XY pair. If your data is waveform in nature, you would store date data in the X values of a wave treated as a waveform. For example:

```
Make/N=100 wave0 = sin(p/8)
SetScale/P x date2secs(2011,4,1), 60*60*24, "dat", wave0
Display wave0
Edit wave0.id; ModifyTable format(wave0.x)=6
```

Here the SetScale command is used to set the X scaling and units of the wave, not the data units as before. In this case, the wave does not need to be double-precision because Igor always calculates X values using double-precision regardless of the wave's data type.

Text Waves

Text waves are just like numeric waves except they contain text rather than numbers. Like numeric waves, text waves can have one to four dimensions.

To create a text wave:

- Type anything but a number into the first unused cell of a table.
- Import data from a delimited text file that contains nonnumeric columns.
- Use the **Make** operation with the /T flag.

You can use the Make Waves dialog to generate text waves by choosing Text from the Type pop-up menu. Most often you will create text waves by entering text in a table. See **Using a Table to Create New Waves** on page II-239 for more information.

You can store anything in an element of a text wave. There is no length limit. You can edit text waves in a table or assign values to the elements of a text wave using a wave assignment statement.

You can use text waves in category plots, to automatically label individual data points in a graph (use markers mode and choose a text wave via the marker pop-up menu) and for storing notes in a table. Programmers may find that text waves are handy for storing a collection of diverse data, such as inputs to or outputs from a complex Igor procedure.

Here is how you can create and initialize text waves on the command line:

```
Make/T textWave= {"First element", "Second and last element"}
```

To see the text wave, create a table:

```
Edit textWave
```

Now you can try some wave assignment statements and see the result in the table:

```
textWave[2] = {"Third element"} // Create new row
textWave += "*" // Append asterisk to each point
textWave = "*" + textWave // Prepend asterisk to each point
```