Version 7

Getting Started Manual

OriginLab Corporation

OriginLab Corporation
One Roundhouse Plaza
Northampton, MA 01060
USA
(413) 586-2013
(800) 969-7720
Fax (413) 585-0126
www.OriginLab.com

Contents

Chapter 1, Introduction

Welcome to Origin

Thank you for purchasing Origin version 7! This manual is provided to familiarize you with the fundamentals of Origin in a minimal amount of time. The manual provides information for new and upgrade users, including:

- A summary of the major new features in version 7.
- An overview of the major Origin concepts and terminology.
- Tutorials covering a broad range of Origin topics.

For additional help using Origin, review the Origin Help file (**Help:Origin**) or visit the OriginLab web site at www.OriginLab.com.

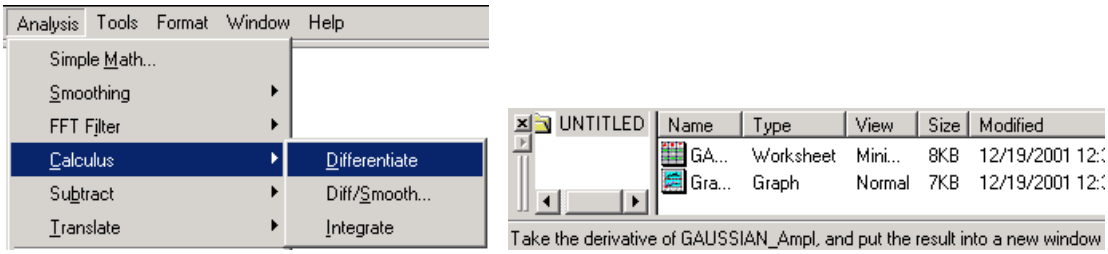
Getting Help Using Origin

If you have a question about using Origin, assistance is available from several different sources.

From the Software

- The status bar in the Origin window provides text clarifying the function of toolbar buttons, tool elements, and menu commands. It also displays Origin status messages.

Figure 1: The Status Bar Messages



- The Origin Help contains information on all of Origin's features. To open the Origin Help, select **Help:Origin** or press F1. If a dialog box is open when you press F1, the Help opens displaying information specific to the dialog box.

Programming Help is also available from the **Help:Programming** submenu. View the section **Introduction to Programming in Origin**, to learn general tips and strategies on programming in Origin. Select **Origin C Reference** to find information on a specific Origin C class or function. Select **LabTalk Language Reference** to find information on the LabTalk programming language.

***Important Note about Origin's Help Files*:** The Origin Help files are compiled HTML Help. To view these Help files, you must have Internet Explorer version 4.0 or higher installed on your computer. (We recommend having Internet Explorer version 5.0 or higher.) Internet Explorer need not be your default browser, but it must be installed.

- Sample Origin projects and data files are provided with Origin. These files are located in the Origin \Samples subfolder. Sample projects show you how to perform analysis routines, create custom graphs, and program routines in Origin.

From the Manuals

- This *Getting Started Manual* includes a "Getting Started Using Origin" section with basic information on using Origin. Tutorials are also provided which step you through common Origin operations.
- The *Programming Guide* provides general tips and strategies on programming in Origin.

From the Web Site

You can access helpful areas of the OriginLab web site by selecting **Help:Origin on the Web**. This menu command opens a submenu providing fast access to a number of useful areas. These resource pages include support, custom tools, the graph gallery, a user forum, and the OriginLab home page. To access the OriginLab home page directly from your browser, go to www.OriginLab.com.

From Your Origin Technical Support Representative

OriginLab and our team of international support representatives are committed to providing high quality technical support to our registered users of Origin. To contact OriginLab Technical Support or to find out how to contact your local support representative, select **Help:Origin on the Web:Technical Support**. Alternatively, go to www.OriginLab.com and click the Technical Support link.

- Customers with local technical support representatives can find contact information on the OriginLab technical support web pages.
- If OriginLab is your technical support representative, you can submit a technical question to OriginLab from the web site.

Additionally, if OriginLab is your technical support representative, you can contact OriginLab Technical Support at tech@originlab.com.

Phone: 1-800-969-7720 or 1-413-586-2013

Additional Products Available from OriginLab

OriginLab provides two major products, Origin and OriginPro. In addition, OriginLab provides custom tools and modules that enhance Origin and OriginPro.

OriginPro

OriginPro includes all the features found in Origin. OriginPro also includes the Peak Fitting Module (see below) and a .GIF export module. OriginPro is an conceived as an application development environment for building custom analysis applications based on Origin. After development, custom applications can be run on the standard Origin version or the OriginPro version.

Create Sophisticated Custom Interfaces

- Create dialog boxes, tabbed tools, and wizards using OriginPro's Dialog Builder.
- Select controls from industry standard development tools.
- Save wizard procedures as a toolbar button.
- Add your own menus and menu commands to the Origin menu bar.

Powerful Programming Environment with Origin C (Origin C is also part of standard Origin)

- ANSI C with some C++ features.
- String, vector, matrix, complex, complex matrix support built-in.
- Access to Origin objects such as worksheets, data plots, and Project Explorer.

- Essential elements of the Numerical Algorithms Group (NAG®) numerical library included for advanced computation.
- Code Builder environment provides syntax coloring, debugging with breakpoints, and output windows.
- Add custom classes into Origin C classes with external DLL. (This feature is only available in OriginPro.)

Design Dynamic Data Exchange (DDE) Applications

- Program your Visual Basic or Visual C++ applications to send data to Origin to display complex graphs in real time.
- Use Origin as a graphics server.

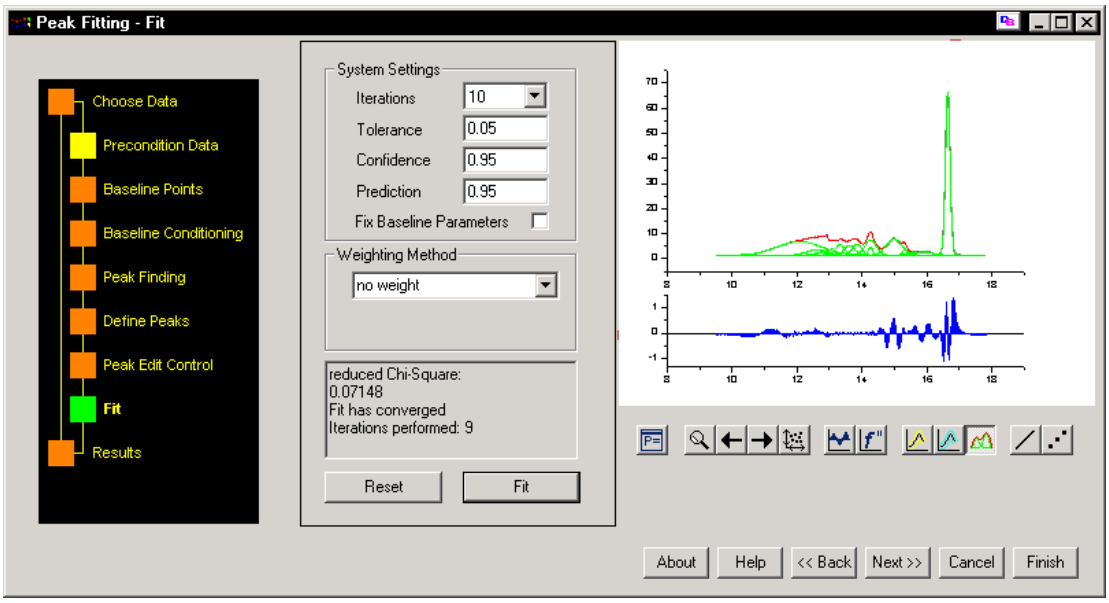
The OriginPro Peak Fitting Module

Work in chromatography, spectroscopy, engineering, pharmacology, and other fields frequently requires analysis of data sets exhibiting multiple peaks. Analysis of multi-peak data is particularly difficult when peaks overlap, or when data are "noisy." The Peak Fitting Module (PFM) provides the tools needed for serious peak analysis, including:

- Data filtering.
- Automatic and/or manual baseline and peak detection.
- Built-in or user-defined curve-fitting functions.
- Highly accurate nonlinear least squares curve fitting.
- Publication-quality output.

The PFM provides a wizard interface to simplify peak analysis.

Figure 2: The Peak Fitting Module



Additional Add-ons

OriginLab offers custom tools and modules that are available from the OriginLab web site (www.OriginLab.com). Some tools are available free of charge and others are available at a cost. The tools add specific enhancements to Origin and OriginPro.

Most of the tools and modules are provided in a special file format with a .OPK extension. After downloading the file, these tools and modules are easily installed by dragging the file from Windows Explorer onto your running copy of Origin or OriginPro.

Chapter 2, What's New in Version 7

Introduction

Origin 7 offers new features that make Origin easier to use and provide increased analysis power. The following sections introduce the major new features in version 7. For more information on a feature, review the Origin Help file (**Help:Origin**). Additionally, review the Release Notes provided with the product.

Ease-of-Use

Annotations

Text Editing

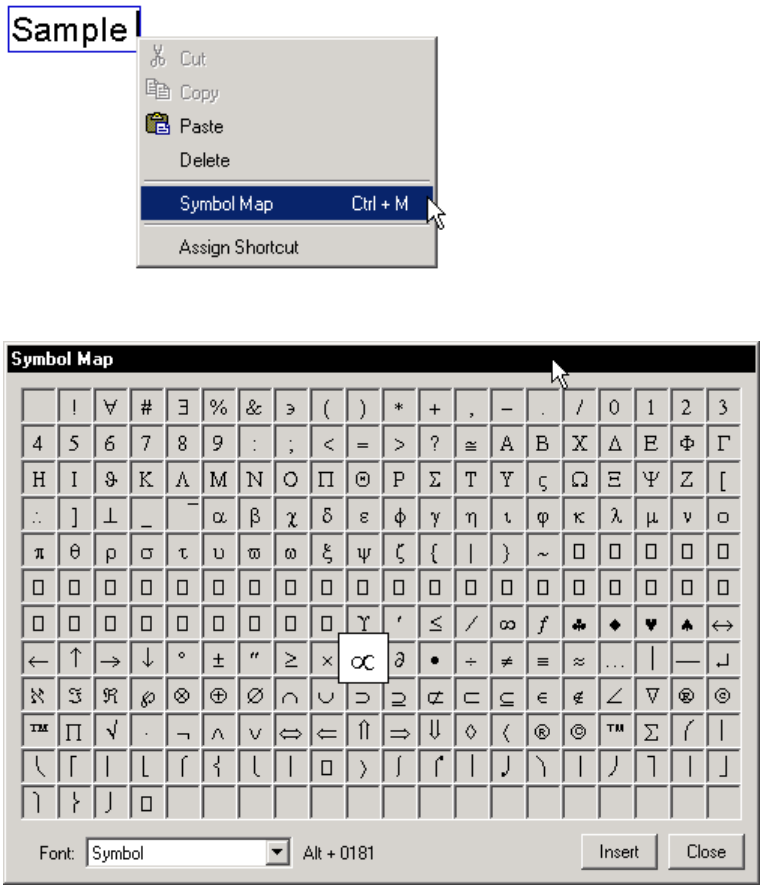
Origin 7 provides enhanced annotation tools including in-place text editing and toolbar button access to common formatting options.

- To create a new text label, right-click and select **Add Text** from the shortcut menu or click the Text Tool button  and then click at the desired location. Then begin typing the text. As you type your text, formatting options are available from the Format toolbar and color control is available from the Style toolbar.

Figure 1: The Format and Style Toolbars

If no text is currently highlighted, the formatting/color option begins at the current cursor location. Otherwise, the formatting/color applies to the highlighted text only. You can also add characters from a selected font set by right-clicking while in in-place editing mode and selecting **Symbol Map** from the shortcut menu (or by pressing CTRL+M).

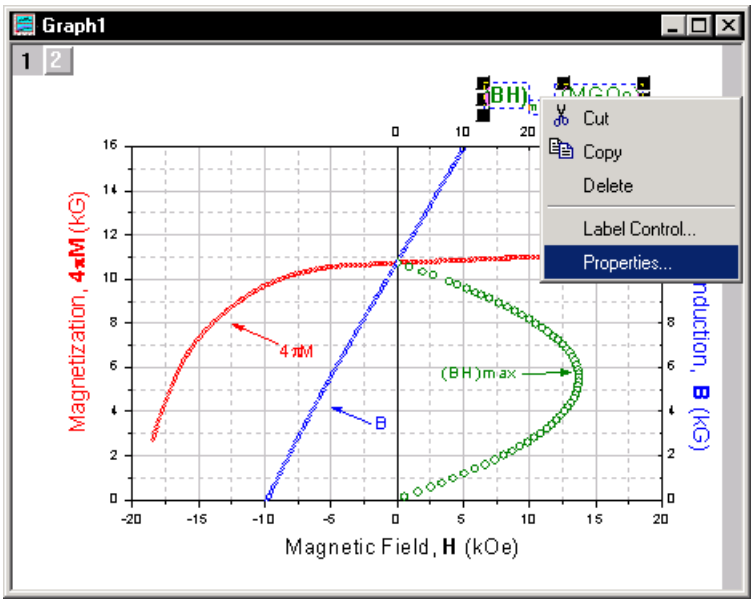
Figure 2: Adding Characters from a Selected Font Set



To exit the text entering and editing mode, click off the label or press ESC.

- To edit an existing text label, double-click to enter the in-place editing mode. (Tip: To temporarily turn off the rotation when you in-place edit rotated labels, select **Tools:Options** to open the Options dialog box. Select the Text Fonts tab and then select the Do Not Rotate Text While In-Place Editing check box.)
- To resize a text label, click once on the label and then select the desired font size from the combo box on the Format toolbar. Alternatively, click the Increase Font or Decrease Font buttons on this toolbar. You can also drag a control handle to resize the label.
- To rotate a text label, click once on the label, pause long enough to avoid a double-click (about a second), and then click a second time on the label. A rotation symbol displays in the middle of the label and rotation handles display at the corners of the label. Click on a rotation handle and rotate the label as desired. (You can also specify a specific rotation angle in the Text Control dialog box.)
- To access the Text Control dialog box, right-click on the text label and select **Properties** from the shortcut menu. Alternatively, press CTRL while double-clicking on the label.

Figure 3: Opening the Text Control Dialog Box



Drawing

Four new drawing tools have been added to the Tools toolbar in Origin 7:

Polygon Tool: To draw a polygon, click on the tool and then click in the window at each of the corner locations for the polygon. Either double-click at the last location or click once and then press ESC.

Region Tool: To draw a region, click on the tool and then click and drag the desired region. Release the mouse button to complete the operation.

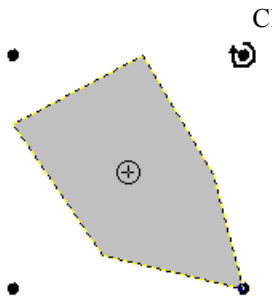
Polyline Tool: To draw a polyline, click on the tool and then click in the window at each of the corner locations for the polyline. Either double-click at the last location or click once and then press ESC.

Freehand Draw Tool: To draw a freehand line, click on the tool and then click and drag the desired line. Release the mouse button to complete the operation.

All of Origin's drawing objects can be rotated and skewed. Additionally, individual points can be moved.

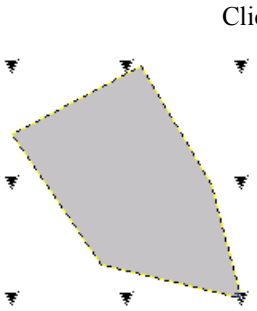
- To rotate an object, click once on the object, pause long enough to avoid a double-click (about a second), and then click a second time on the object. A rotation symbol displays in the middle of the object and rotation handles display at the corners of the object. Click on a rotation handle and rotate the object as desired.

Figure 4: Rotating an Object



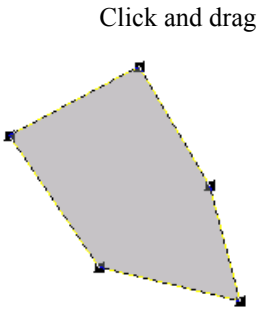
- To skew an object, click once on the object, pause long enough to avoid a double-click (about a second), and then click a second time on the object. Pause again and then click a third time on the object. Triangular skew handles display at the corners of the object. Click on a skew handle and drag as desired.

Figure 5: Skewing an Object



- To move points in an object, follow the "skew" procedure and then click one more time (a total of four clicks with pauses in between). Handles appear on moveable points. Drag the desired points to new locations.

Figure 6: Moving an Object's Points



Additionally, when a drawing object is selected, the Style toolbar buttons are available for customizing the object's display. For closed objects, this includes the pattern and fill color controls.

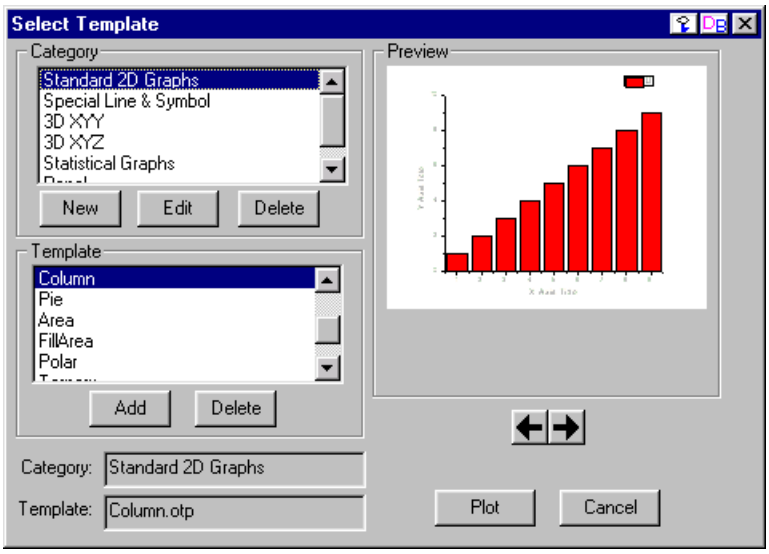
Additional object controls are available from the Object Control dialog box which is accessed by double-clicking on the object.

Plotting

Template Library Tool

Origin provides a Template Library tool for categorizing and accessing graph templates. To open the Template Library tool when a worksheet or an Excel workbook is active, select **Plot:Template Library**. In addition to organizing graph templates, you can also use the tool to plot your worksheet or Excel workbook data. If you highlighted data in the worksheet or workbook before opening the tool, and your data selection is appropriate for the template you've selected, then click the Plot button to plot the data into the template. If you did not highlight data or if your selection was not appropriate for the template you've selected, then click the Plot button to open an intermediary dialog box for data selection.

Figure 7: The Template Library Tool



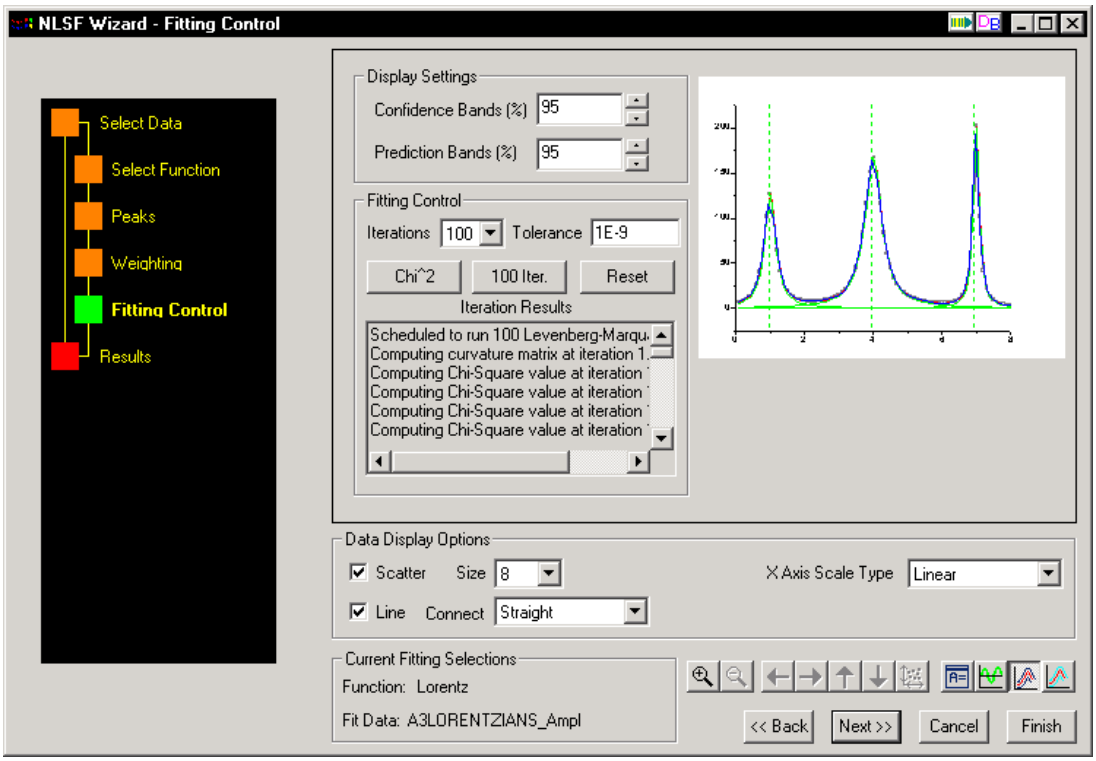
Analysis

NLSF Wizard

Origin 7 provides a wizard for performing nonlinear least squares fitting. The NLSF wizard is easier to use than the advanced fitting tool (NLSF), as it steps you through the fitting process. The wizard provides only the most frequently used fitting options. For complete fitting options, open the NLSF.

To open the NLSF Wizard, select **Analysis:Nonlinear Curve Fit:Fitting Wizard**.

Figure 8: The NLSF Wizard



Analysis Apply Button

The following analysis routines now have an Apply button available in their respective dialog boxes. When the Apply button is clicked, the interim results display in the graph. The dialog box remains open and is available for further changes. Each time you change a dialog box setting and click Apply, the results update in the graph. The results are not finalized until you click OK.

- The following graph menu commands have an Apply button:

Analysis:Smoothing:Savitzky-Golay, **Adjacent Averaging**, and **FFT Filter**

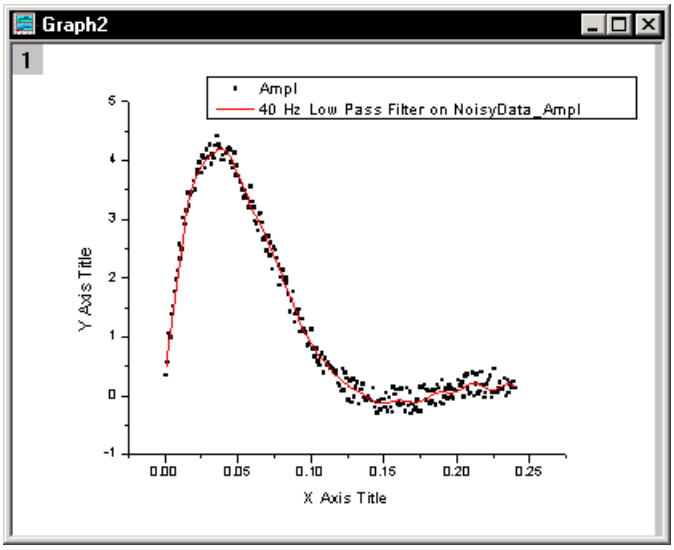
Analysis:FFT Filter:Low Pass, **High Pass**, **Band Pass**, and **Band Block**

Analysis:Interpolate/Extrapolate

- The following worksheet menu commands have an Apply button:

Statistics:Descriptive Statistics:Frequency Count

Figure 9: 40 Hz Low Pass Filter Applied



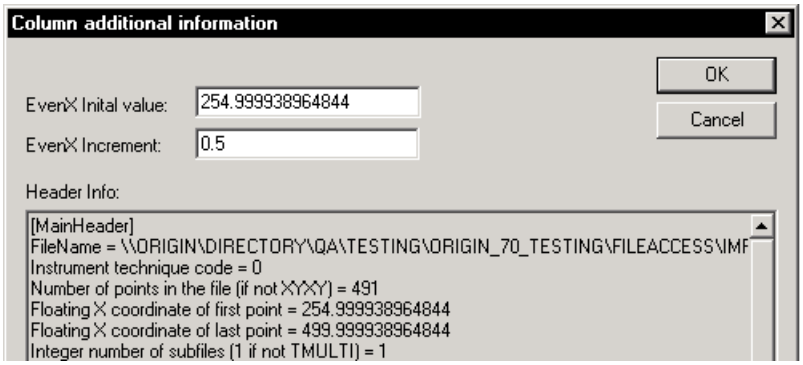
Data Import and Handling

Thermo Galactic SPC

You can now import Thermo Galactic SPC data files into Origin by selecting **File:Import:Galactic (SPC)**. Origin supports both single and multiple arrays.

- If the data file contains an array of X values, then Origin assigns those values to an X column.
- If the initial X value and the increment are stored in the header, then Origin creates a hidden X column with the correct starting value and increment. To view this information in Origin, perform one of the following operations:
    - Double-click on the black triangle located in the upper-left corner of the column heading. This action opens the Column Additional Information dialog box. You can modify the starting X value and increment in the associated text boxes.

Figure 10: Reviewing the Starting X Value and Increment



- Click on the column heading to select the column and then select **Format:Set Worksheet X**. This menu command also opens a dialog box for modifying the starting X value and increment.

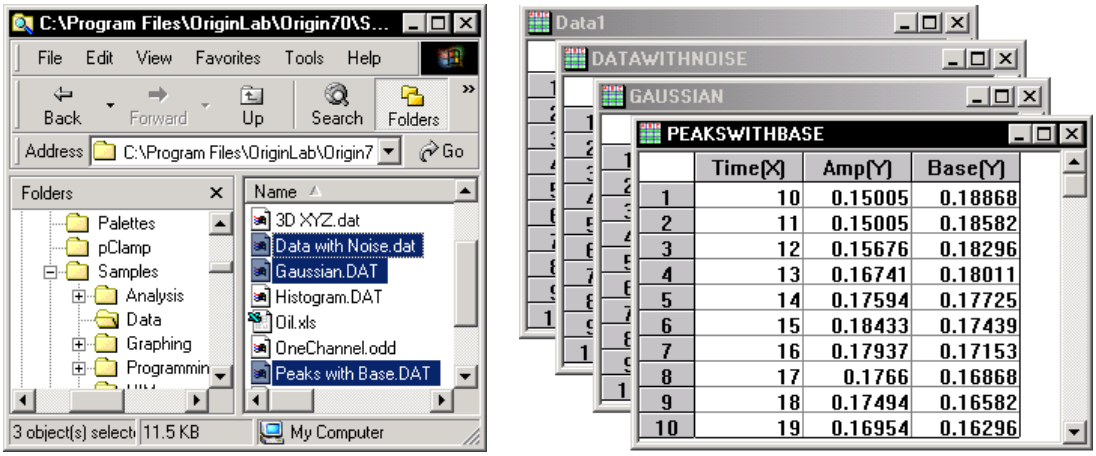
To view the hidden X column, select **View:Show X Column**.

Drag-and-Drop

You can drag-and-drop ASCII, SigmaPlot, Minitab, and Thermo Galactic SPC files into Origin. Once you have selected the file in Windows Explorer, if Origin isn't currently open you can drag the file onto your Origin desktop icon. If Origin is already open, you can drag the file over the Origin taskbar button and hold there until Origin becomes active. Then continue dragging and drop the file into the Origin workspace.

You can drop the data files into existing worksheets or graphs, or you can drop into a blank location of the workspace to import into a new worksheet for a single file, or multiple worksheets for multiple files.

Figure 11: Dragging Data Files Into Origin

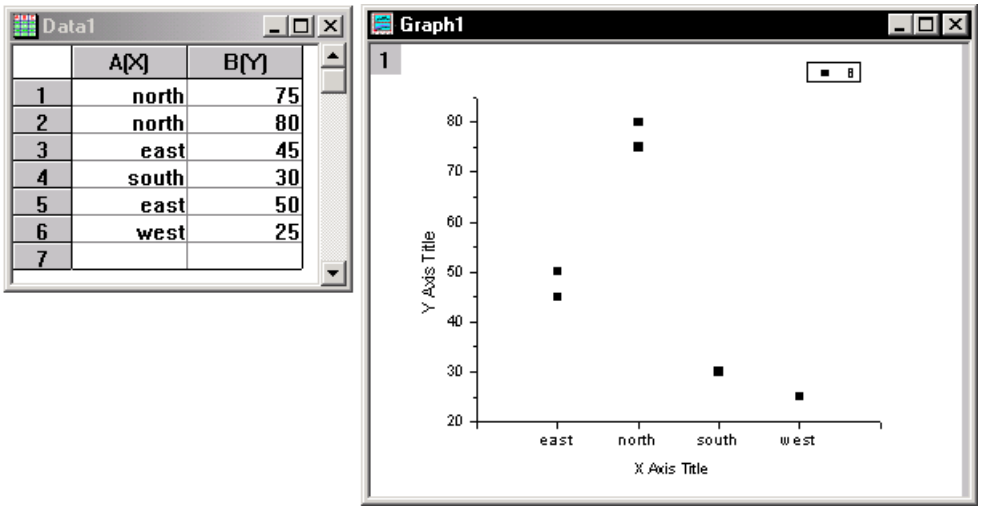


Categorical Data Support

Origin 7 supports plotting categorical data in both X and Y columns. Before plotting categorical data, you must set the column to Categorical by highlighting the column and selecting **Column:Set as Categorical**.

- When you plot a Categorical X column and one or more associated Y columns, Origin creates a graph with the X categories as X axis tick labels. These tick labels are organized alphabetically (categories starting with numeric values are first) and then evenly spaced across the axis. The Y data is plotted using the associated X tick values.

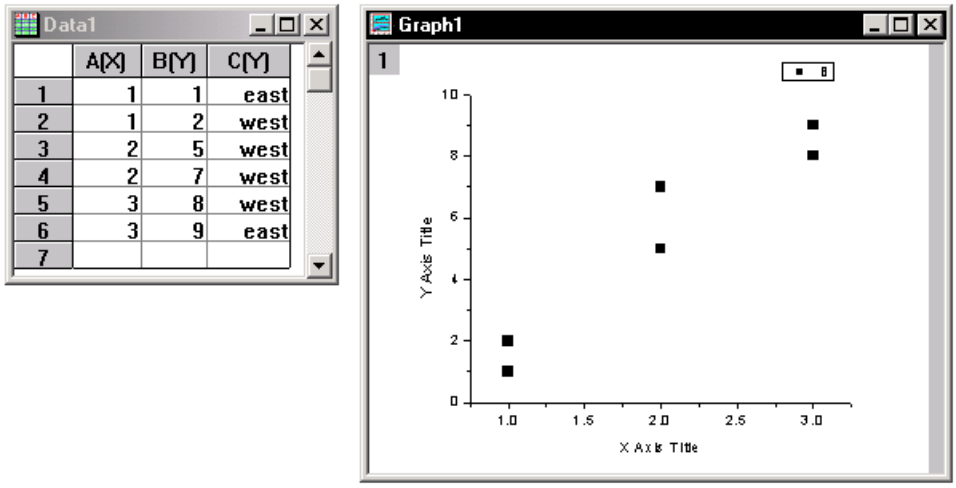
Figure 12: Plotting Categorical X Data



- If your worksheet contains a Categorical Y column, then you can map this categorical data to your data plots, displaying categories of data using the same symbol shape, color, size, or other plot attribute.

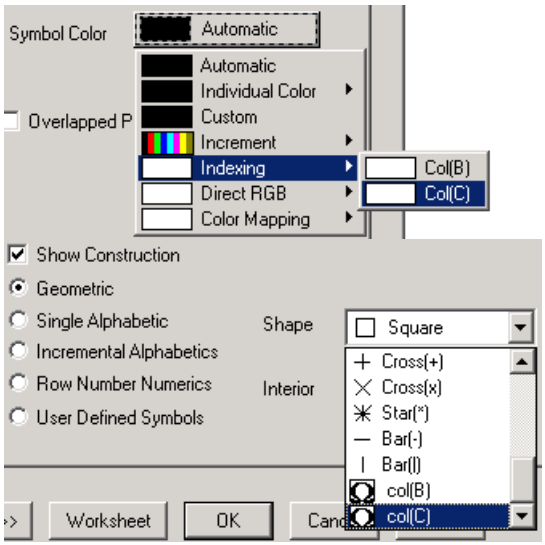
For example, in the following figure, the A(X) and B(Y) columns are plotted using the Scatter template.

Figure 13: Plotting the A(X) and B(Y) Columns



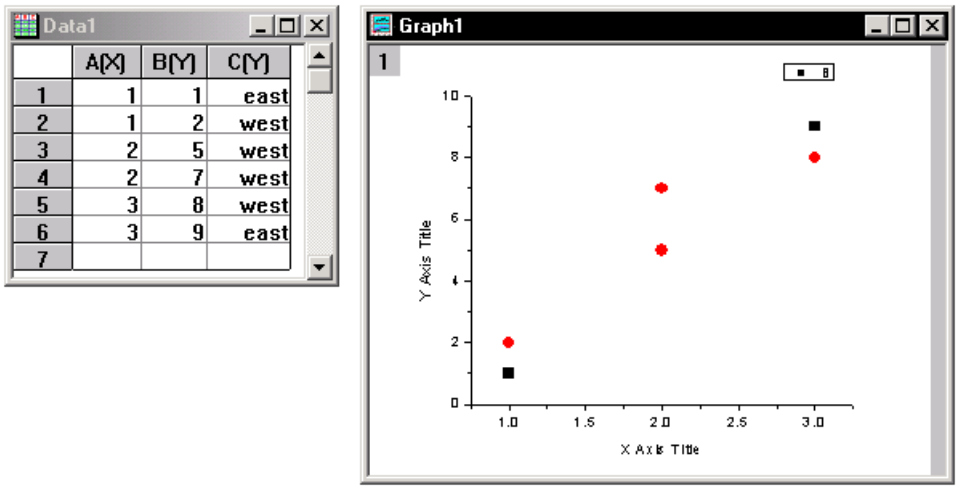
To display categories of data (east, west) using the same symbol shape and color, open the Symbol tab of the Plot Details dialog box and edit the Symbol Color and Shape drop-down lists as shown in the following figure. In this example the colors for each category will be *indexed* from the color list.

Figure 14: Mapping the Symbol Color and Shape to Column C



The resultant graph displays the data using the column C categories for both the symbol color and shape. To do this, Origin alphabetizes the categories (categories starting with numeric values are first). Because color indexing was selected, Origin assigns the first category the first color in the color list, the second category the second color, etc. Origin performs this same alphabetic assignment for all other mapped plot attributes.

Figure 15: The Resultant Graph



Analysis Power

New Graph Types

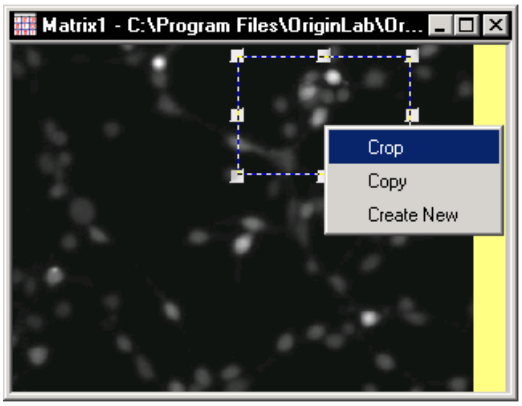
Image Graph

Origin 7 provides enhanced support for importing, viewing, and plotting raster graphic images. To import a gray scale, 8-bit color or higher resolution color image into the active matrix, select **File:Import Image**.

When you first import the image, Origin displays a device independent bitmap (DIB) of the image in the matrix.

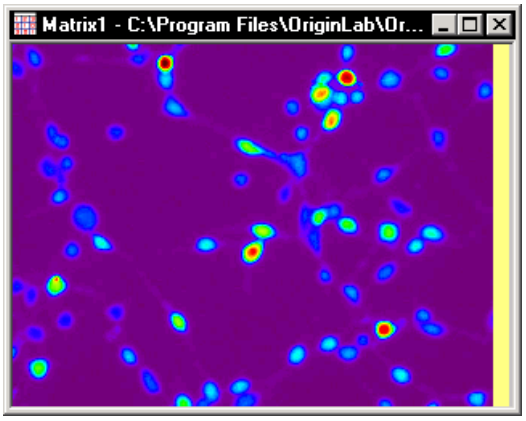
If you are only interested in a region of the image, you can select a region of the DIB using the Rectangle Tool (in "region of interest mode")  on the Tools toolbar.

Figure 16: Selecting a Region of Interest



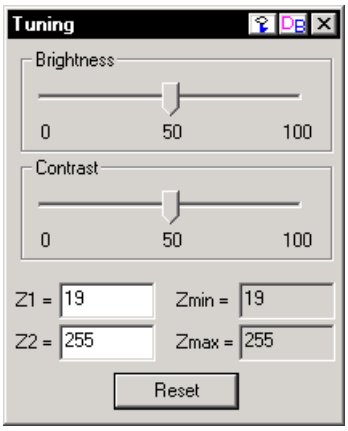
You can also view the image using a built-in or user supplied color palette. Viewing the image using a specified color palette may clarify regions of the image. To view the image using a color palette, you must first convert the DIB to matrix data. To do this, select **Image:Convert to Gray + Data**. Origin converts each pixel to an RGB value and then assigns the corresponding matrix cell an index number to a gray scale palette, based on the RGB value of the pixel. To display the image using a palette other than gray scale, select **Image:Palette:*PaletteSelection***.

Figure 17: Viewing the Image Using a Built-in Palette for Improved Clarity



When viewing the image from a palette, Origin maps each cell's index value to a color in the selected palette. Thus, the image's full matrix Z value range is mapped to the palette. You can adjust the brightness and the contrast of the image using the Tuning tool. To open this tool, select **Image:Tuning**. When you adjust the Contrast slider, you are increasing or decreasing the Z value range that is mapped to the palette. When you adjust the Brightness slider, you are shifting the range of Z values that are mapped to the palette.

Figure 18: Adjusting the Brightness and Contrast of the Image



To plot the image into a graph window, select **Plot:Image Plot**.

Figure 19: Plotting the Image into a Graph

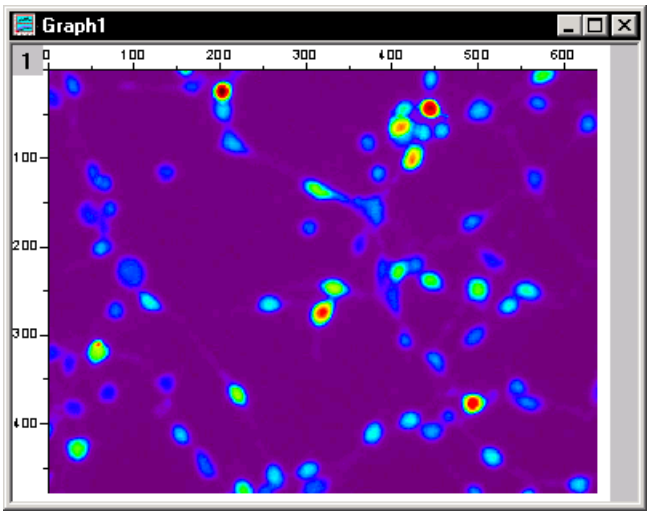


Image Histogram

After importing a raster graphic image into a matrix, Origin can create a histogram of the intensity values in the image. To plot a histogram from the image in the matrix, select **Plot:Histogram**.

Figure 20: Example Image Histogram

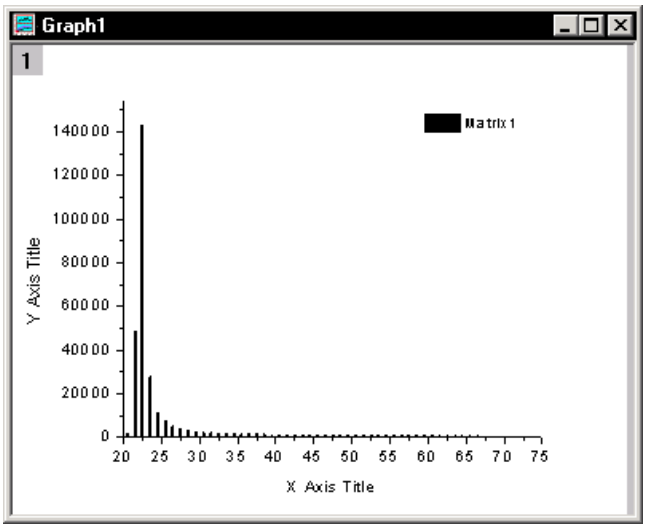
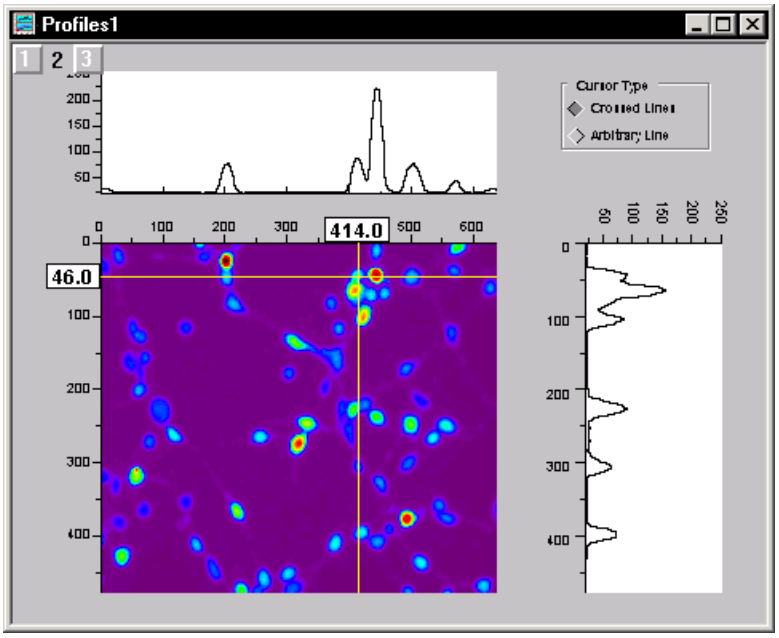


Image Profiling

Matrix images can also be plotted using a graph template that includes X and Y projections. To plot to this template, select **Plot:Profiles**. You can drag the lines to view different X and Y projections. You can also view the projections using an arbitrary line.

Figure 21: Viewing the Images X and Y Projections

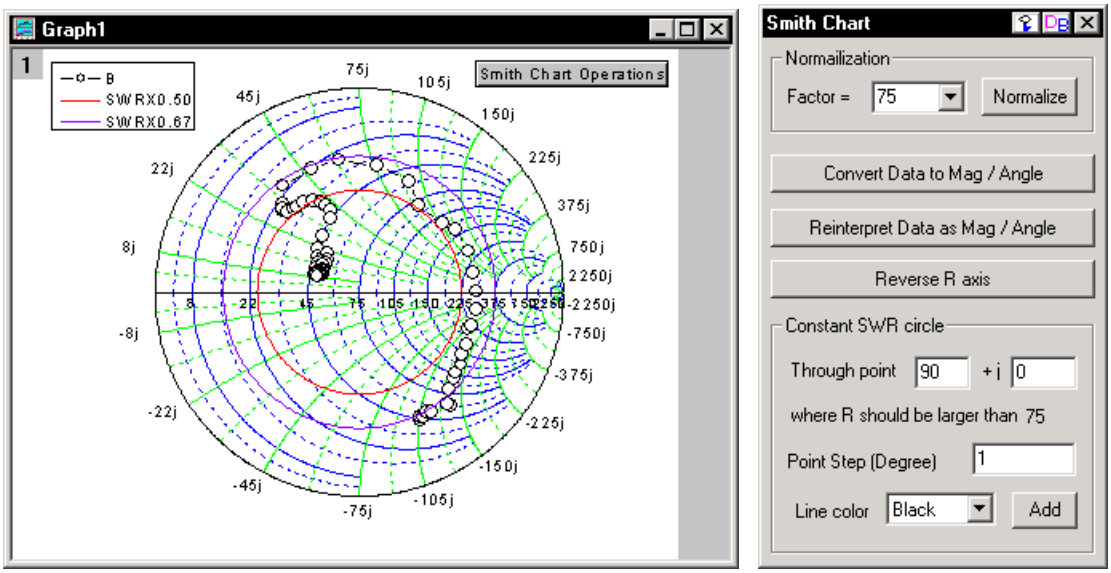


Smith Chart

You can now create Smith® Charts in Origin 7. To plot data using the Smith Chart template, select **Plot:Smith Chart** or click the Smith Chart button  on the 2D Graphs toolbar.

To customize the Smith Chart, edit the Plot Details and Axes dialog boxes. Additionally, click the Smith Chart Operations button to open the Smith Chart tool.

Figure 22: Smith Chart with Operations Tool



Statistical Analysis

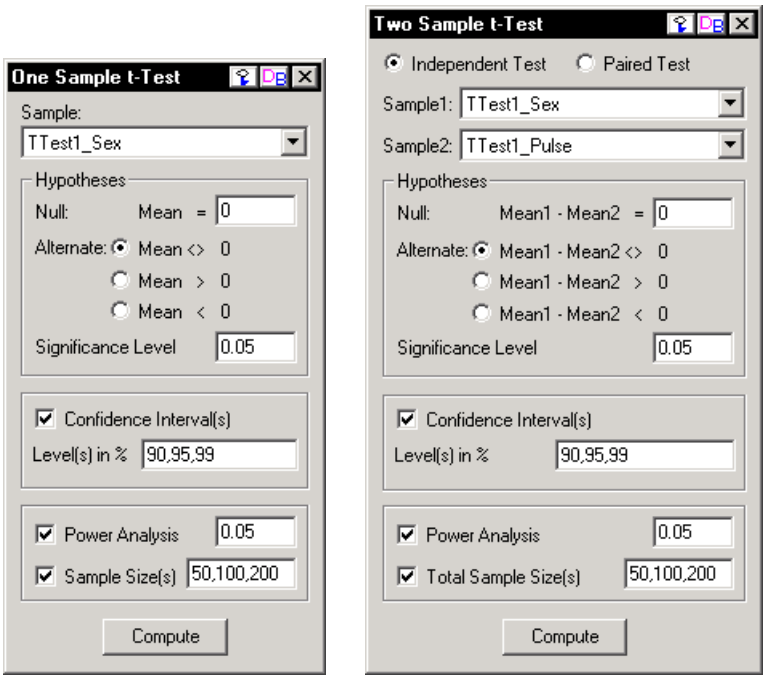
One / Two Sample t-Test

One and two sample t-Tests have been completely redesigned and expanded in Origin 7. The following new features have been added:

- You can now select new data sets, change settings, and re-compute without having to re-open the dialog box each time.
- Both one and two tailed tests can now be computed by selecting any one of three Alternate Hypotheses.
- Confidence intervals for a number of different confidence levels can now be computed.
- Actual Power can now be computed for any specified alpha level.
- Hypothetical Power for a number of different sample sizes can now be computed.

One and two sample t-Tests are available from the **Statistics:Hypothesis Testing** menu.

Figure 23: One and Two Sample t-Test Dialog Boxes



Survival Analysis

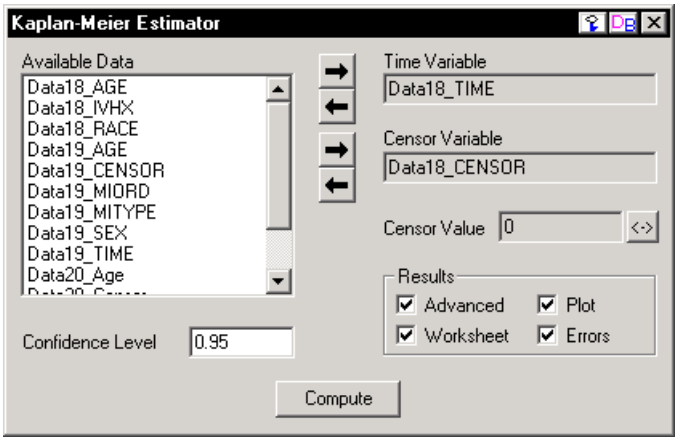
The Survival Analysis features are new in Origin 7. Two computations are available:

- Kaplan-Meier Product Limit Estimator
- Cox Proportional Hazards Model

Both of these computations are used to estimate the survivorship function which is the probability of survival to a given time based on a sample of failure times.

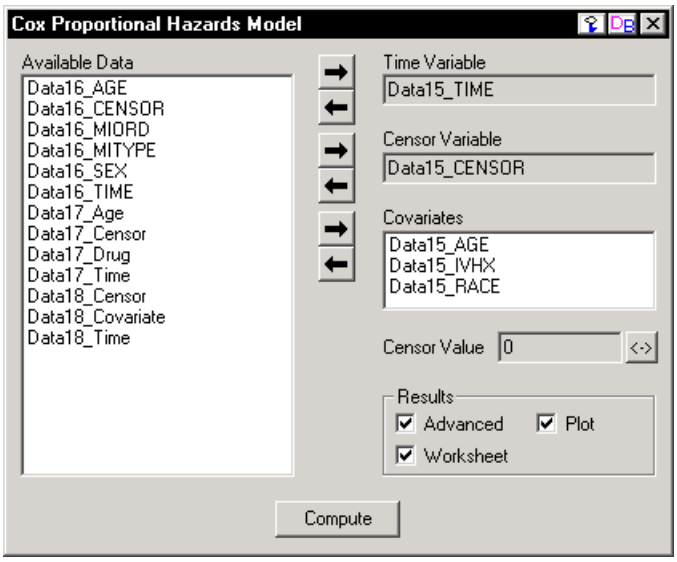
To use the Kaplan-Meier estimator, select **Statistics:Survival Analysis:Kaplan-Meier Estimator**.

Figure 24: Kaplan-Meier Estimator Dialog Box



To use the Cox Proportional Hazards model, select **Statistics:Survival Analysis:Cox Proportional Hazards Model**.

Figure 25: Cox Proportional Hazards Model Dialog Box



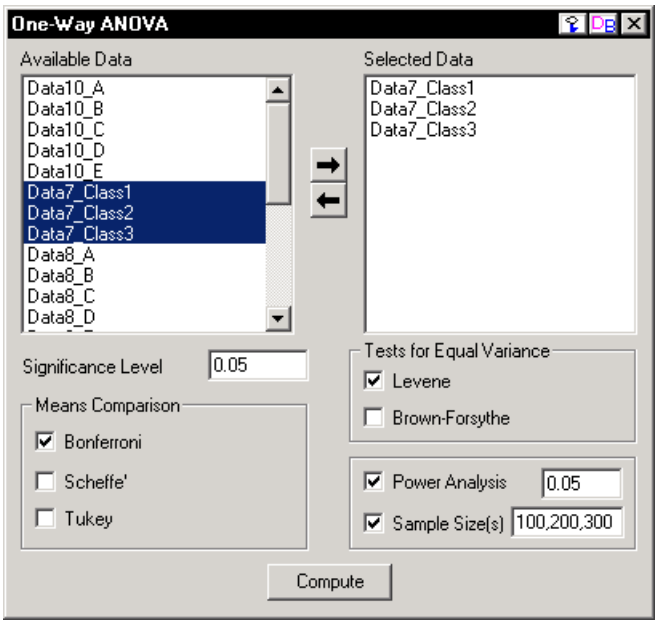
One-Way ANOVA

One-way ANOVA has been redesigned and expanded in Origin 7. The following new features have been added:

- You can now select new data sets, change settings, and re-compute without having to re-open the dialog box each time.
- Non-contiguous column selection from any worksheet in the project.
- Three different methods of Means Comparison (Bonferroni, Scheffé, Tukey) can now be computed.
- Two different Tests for Equal Variance (Levene, Brown-Forsythe) can now be computed.
- Actual Power can now be computed for any specified alpha level.
- Hypothetical Power for a number of different sample sizes can now be computed.

One-way ANOVA is available from the **Statistics:ANOVA** menu.

Figure 26: One-way ANOVA Dialog Box



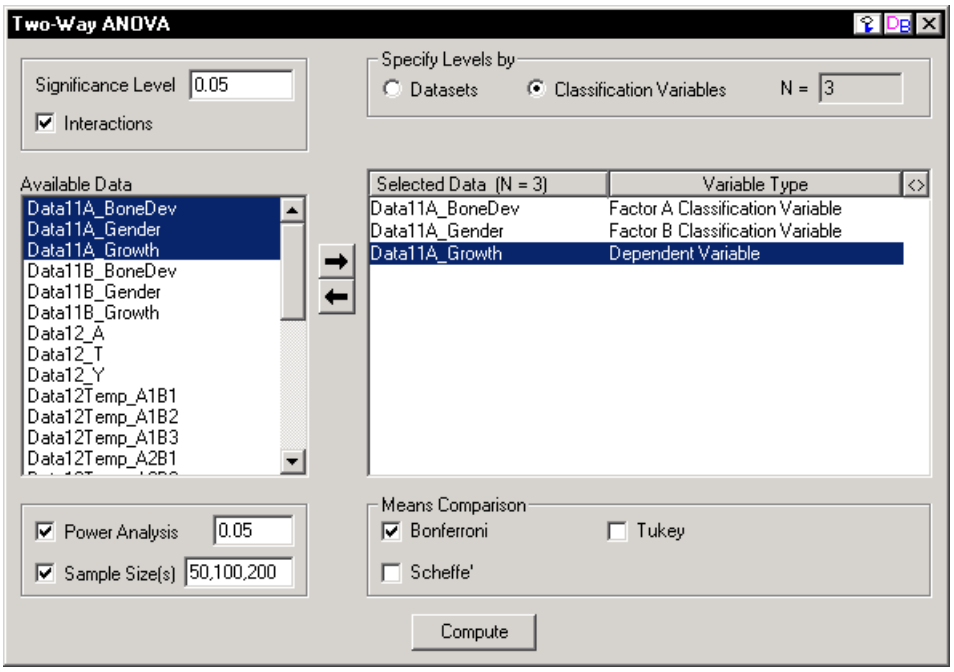
Two-Way ANOVA

Two-way ANOVA evaluates the effect of two independent factors on a measured response and whether or not there is an interaction between the two factors. This feature is new in Origin 7 and will support the following new computations:

- You can select new data sets, change settings, and re-compute without having to re-open the dialog box each time.
- Non-contiguous column selection from any worksheet in the project, and the ability to group levels of each factor either by classification variables or by data set.
- Three different methods of Means Comparison (Bonferroni, Scheffé, Tukey) can be computed.
- A computation that determines whether or not there are any interactions between the two factors.
- Actual Power can be computed for any specified alpha level.
- Hypothetical Power for a multiple number of different sample sizes can be computed.

Two-way ANOVA is available from the **Statistics:ANOVA** menu.

Figure 27: Two-way ANOVA Dialog Box



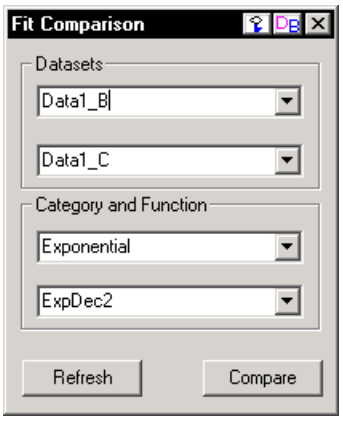
Normality Test

You can now perform a Shapiro-Wilk normality test by selecting one or more columns of data and then selecting **Statistics:Descriptive Statistics:Normality Test (Shapiro-Wilk)**. This test detects departures from normality without requiring that the mean or variance of the hypothesized normal distribution be specified in advance. For each selected data set, the sample size N, the Shapiro-Wilk statistic W and its significance level for testing normality P(W), and the decision rule are output to the Results Log.

Fit Comparison

A new Fit Comparison tool is available by selecting **Tools:Fit Comparison**. This tool compares two data sets by fitting the same function to the data. It then uses an F-test to determine whether the two data sets are significantly different from each other. The results are output to the Results Log.

Figure 28: Fit Comparison Tool



Programming

Origin C

Origin 7 introduces a new programming language called Origin C. Origin C supports a nearly complete ANSI C language syntax and a subset of C++ features including internal and DLL-extended classes. Furthermore, Origin C is "Origin aware". This means that Origin objects such as worksheets and graphs are mapped in Origin C, allowing direct manipulation of these objects and their properties from Origin C.

Typical programming routines in Origin include the following:

- Adding functionality to Origin by creating new importing, analysis, graphing, and exporting routines.
- Automating the work you do in Origin.
- Performing simulations in Origin, with live feedback.

To learn more about programming using Origin C, select **Help:Programming:Program Guide** from the Origin menu. Additionally, sample Origin projects and associated source files are included in the \Samples\Programming subfolders.

Code Builder

Code Builder is Origin's integrated development environment. To open Code Builder, click the Code Builder button  on the Standard toolbar. Code Builder provides standard tools for writing, compiling, and debugging your Origin C functions. Once an Origin C function is compiled, the function becomes accessible from Origin.

NAG Numerical Library

Origin 7 includes the following Numerical Algorithms Group (NAG®) function libraries:

a02 - Complex Arithmetic
c06 - Fourier Transforms
e01 - Interpolation
e02 - Curve and Surface Fitting
F - Linear Algebra
f06 - Linear Algebra Support Functions
g01 - Simple Calculations on Statistical Data
g02 - Correlation and Regression Analysis
g03 - Multivariate Methods
g04 - Analysis of Variance
g08 - Nonparametric Statistics
g11 - Contingency Table Analysis
g12 - Survival Analysis
s - Approximations of Special Functions

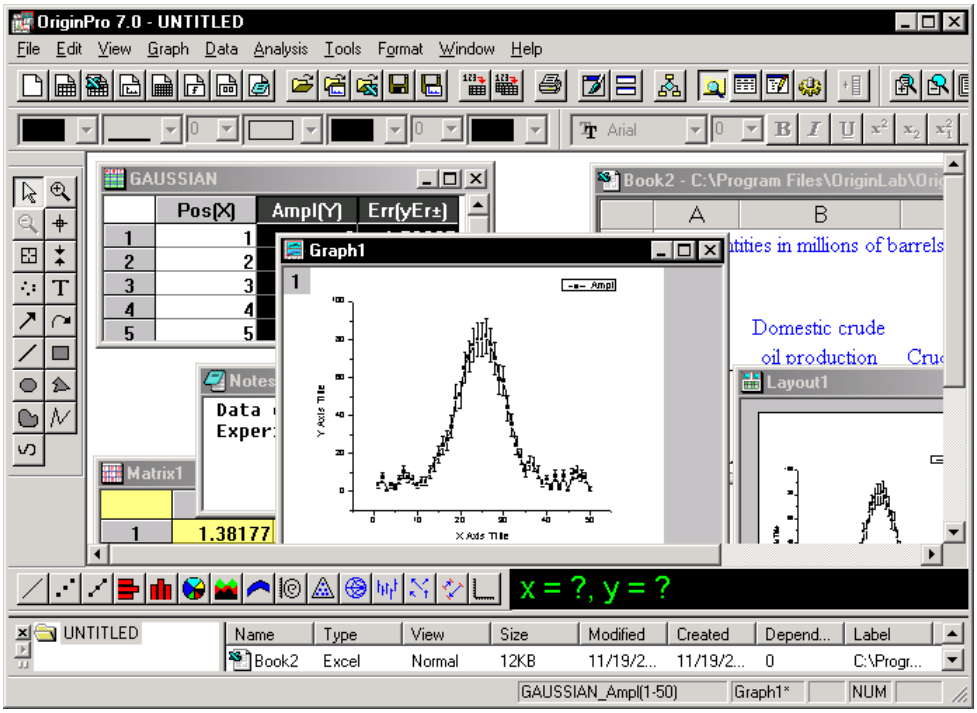
Many of these functions are called from built-in Origin routines. However, you can also call any of these NAG functions from Origin C. To learn more, review the sample Origin project files provided in the \Samples\Programming\NAG ... folders.

Chapter 3, Getting Started Using Origin

The Origin Workspace

When you start Origin, a new project opens displaying a worksheet window in the workspace. The worksheet is one type of window available in Origin. Origin also provides graph (including function graph), layout page, Excel workbook, matrix, and notes windows. Having various windows allows you to simultaneously view different visual representations of your data - such as data in a worksheet versus a graph - simplifying data manipulation and analysis.

Figure 1: The Origin Workspace and Supported Window Types



Menus and Menu Commands

Origin's menu bar provides commands to perform operations on the active window and to perform general operations such as opening a Help file or turning on the display of a toolbar. The menu bar changes as you change the active window. For example, the following figures compare the worksheet and graph menu bars.

Figure 2: The Worksheet Window Menu Bar

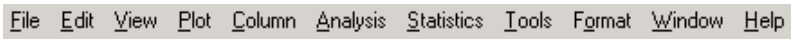
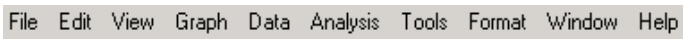
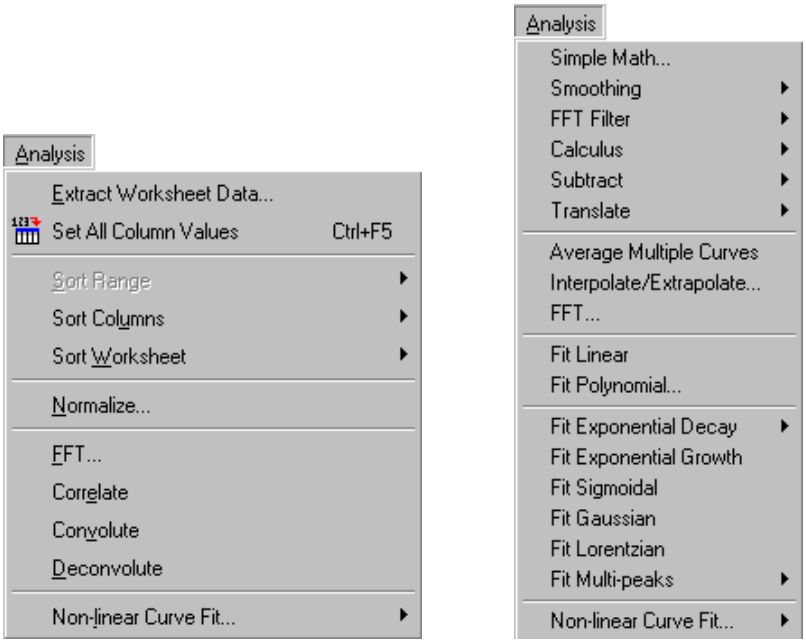


Figure 3: The Graph Window Menu Bar



Menus are also sensitive to the active window. For example, the following figure compares the worksheet and graph **Analysis** menus.

Figure 4: The Worksheet and Graph Analysis Menus



Origin provides two menu "levels" which determine the number of menu commands that are available. By default, Origin displays the "full menu", which means that all available menu commands are provided. However, Origin also offers a "short menu" level, which provides a reduced set of menu commands for performing basic operations only. To activate this reduced set of commands, select **Format:Menu:Short Menus**. At any time you can re-activate the full set of commands by selecting **Format:Menu:Full Menus**.

Some menu commands have shortcut keys associated with them. If available, the shortcut key displays to the right of the menu command. For example, when a worksheet window is active, you can press CTRL+F5 to access **Set All Column Values**. (Note: You can't use a shortcut key if the menu is open.)

Some menu commands also have bitmaps that display to the left of the command. The bitmap indicates that the menu command also has toolbar button access. To access the command from a toolbar, look for the toolbar button represented by the command's bitmap.

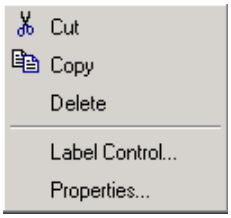
Figure 5: Accessing a Command from a Toolbar

(To learn how to open additional toolbars, such as the Worksheet Data toolbar, see "Toolbars" on page 30.)

To turn off the display of bitmaps in the menus, select **Tools:Options** to open the Options dialog box. Select the Miscellaneous tab and then clear the Display Bitmaps in Menus check box. After you click OK, you are asked if you want to save this setting for future Origin sessions.

Many commands are also available from shortcut menus. To open a shortcut menu, right-click on the object you want to perform an action on. For example, if you right-click on a text label, the shortcut menu in the following figure opens.

Figure 6: Opening a Shortcut Menu



Toolbars

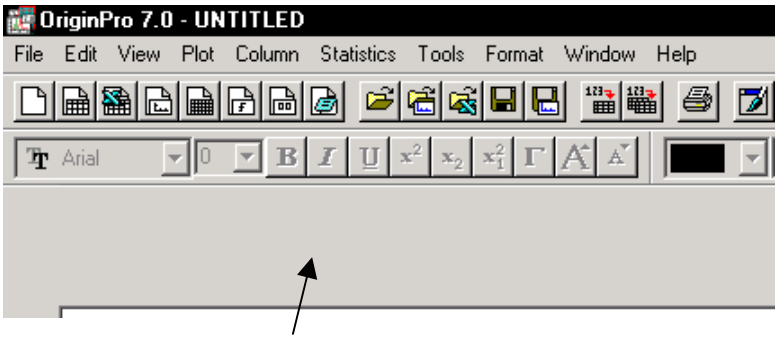
Origin provides toolbar buttons for frequently used menu commands. As with menu commands, some toolbars are only available when a particular window (for example, a worksheet) is active. Additionally, a toolbar that is available for multiple window types may contain buttons that are window-sensitive.

When you position the mouse pointer over a toolbar button, a view box opens displaying the button name, which indicates its purpose. A more detailed description also displays in the status bar.

Figure 7: Viewing a Button's Name and Purpose

If you open Excel workbooks in Origin, when you change the active window from an Excel workbook to any other window type (for example, a worksheet), or when you close an Excel workbook, the toolbar region displays a blank area where the Excel toolbars were located (see the following figure).

Figure 8: Blank Area in the Toolbar Region



This area is called a toolbar spacer. To hide the toolbar spacer, right-click in the region and select **Hide Toolbar Spacer** from the shortcut menu. When you re-activate the Excel workbook window or re-open a

workbook, Origin will automatically show the toolbar spacer with the Excel toolbars. (To prevent Origin from using the toolbar spacer, select **Tools:Options** to open the Options dialog box. Select the Miscellaneous tab and then clear the Use Toolbar Spacer check box. After you click OK, you are asked if you want to save this setting for future Origin sessions.)

Note: To open the Options dialog box when an Excel workbook is active, select **Window:Origin Options**.

When you first start Origin, the following toolbars are available: Standard, Graph, Format, Style, Tools, and 2D Graphs.

Figure 9: The Standard Toolbar

The Standard toolbar provides buttons for opening, saving, and creating new projects and windows, and for importing ASCII data. It also provides buttons for general window operations such as printing, duplicating, and refreshing windows. The Standard toolbar provides buttons for opening Project Explorer, the Results Log, the Script window, and Code Builder. A button is provided for custom programming. A button is also provided for adding a column to the worksheet.

Figure 10: The Graph Toolbar

The Graph toolbar is available when a graph or layout page is active. It provides buttons to zoom in and out and to rescale axes to show all the data. It provides buttons to display data plots in multiple layers, display layers in multiple windows, and to merge windows. Labeling buttons are available for legends and a time/date stamp.

Figure 11: The Format Toolbar

The Format toolbar is available when a text label is active. This toolbar provides text formatting buttons. Color control is available from the Style toolbar.

Note: The Greek button  uses the Symbol font set. To associate the button with a different font set, select **Tools:Options** to open the Options dialog box. Select the Text Fonts tab and then select the desired font set from the Greek drop-down list.

Figure 12: The Style Toolbar

The Style toolbar is available when a text label or other annotation is selected. It provides buttons to set the line and fill color, style, and point size.

Figure 13: The Tools Toolbar

The Tools toolbar provides text, arrow, line, and other annotation buttons. It also provides buttons to enlarge a region of a graph. The Tools toolbar also provides buttons to read the XY (and Z, if 3D or contour) location on the page , and the XY (and Z) location of a data point . You can also define a range of data . Furthermore, a button is provided to draw a data plot .

For more information on the Screen Reader, Data Reader, and Data Marker buttons, see "Tutorial 2, Exploring Your Data".

Note: If you are viewing an image in a matrix, you can display the Rectangle tool in the "region of interest" mode . The region of interest mode allows you to select a region of the image to crop, copy, or duplicate. The region of interest mode is controlled from the **Tools:Show Tools as ROI** menu command.

Figure 14: The 2D Graphs Toolbar

The 2D Graphs toolbar is available when a worksheet, Excel workbook, or graph window is active. It provides buttons for the common 2D graph templates, and for accessing a custom graph template.

- When a worksheet or Excel workbook is active, first select the data that you want to plot and then click the button on the 2D Graphs toolbar to plot your data.
- When a graph window is active, you can change the graph type (for example, from scatter to column) for the active data plot. To activate a data plot, select the data plot from the **Data** menu. Then click the button on the 2D Graphs toolbar to change the graph type.

In addition to these default toolbars, you can open the following additional toolbars. To do this, select **View:Toolbars** to open the Customize Toolbar dialog box. Then select the desired toolbar from the Toolbars tab.

Figure 15: The Edit Toolbar

The Edit toolbar provides cut, copy, and paste buttons.

Figure 16: The 2D Graphs Extended Toolbar

The 2D Graphs Extended toolbar is available when a worksheet, Excel workbook, or graph window is active. It provides buttons for the additional 2D graph templates not accessible from the 2D Graphs toolbar.

Figure 17: The 3D Graphs Toolbar

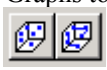
The 3D Graphs toolbar is available when a worksheet, Excel workbook, or matrix is active. The first two buttons  are available for plotting XYZ data. The next four buttons  are available for plotting XYY data. The remaining buttons  are available for plotting matrices. The last button, the Image Plot button  , is available for plotting images.

Figure 18: The 3D Rotation Toolbar

The 3D Rotation toolbar is available when a 3D graph is active. This toolbar provides buttons to rotate the graph and change the perspective.

Figure 19: The Worksheet Data Toolbar

The Worksheet Data toolbar is available when a worksheet is active. This toolbar provides buttons to perform statistics on columns or rows of data, sort, use functions to set column values or mathematically transform values, update these column values, and fill columns with row numbers or random numbers.

Figure 20: The Column Toolbar

The Column toolbar is available when a worksheet column is selected. It provides buttons to set the column plotting designation and to move columns.

Figure 21: The Layout Toolbar

The Layout toolbar is available when a layout page is active. It provides buttons to add pictures of graphs or worksheets to the layout page.

Figure 22: The Mask Toolbar

The Mask toolbar is available when a worksheet or graph is active. This toolbar provides buttons to mask data from analysis.

Figure 23: The Object Edit Toolbar

The Object Edit toolbar is available when one or more annotation objects are selected in the active window, or when more than one picture is selected in a layout page. The toolbar provides buttons to align the selected objects/pictures, and to change the drawing order of the objects/pictures. Buttons are provided to group objects so that you can move or align the group. Additionally, buttons are provided to change the drawing order of objects relative to data plots.

Figure 24: The Arrow Toolbar

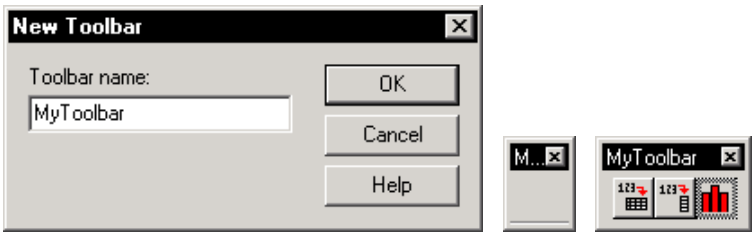
The Arrow toolbar is available when one or more lines or arrows are selected. This toolbar provides buttons to align multiple lines/arrows, and to customize the arrow head.

In addition to adding Origin's default toolbars to your workspace, you can add and remove buttons from any of the default toolbars, or create new toolbars containing a single button or a combination of buttons from the default toolbars.

To add and remove buttons from any of Origin's default toolbars, select **View:Toolbars** to enter the toolbar editing mode. You can now drag toolbar buttons between toolbars. To remove a button from a toolbar, drag the button off the toolbar (not on to another toolbar) and release the mouse button. To add a button to a toolbar, select the Button Groups tab on the Customize Toolbar dialog box. From the Groups list box, select the toolbar which displays the button by default. Then drag the desired button onto any of the toolbars.

To create a new toolbar, you can drag a button into the workspace or into the toolbar region (not on to another toolbar). Alternatively, click New on the Toolbars tab of the Customize Toolbar dialog box. Type the desired name in the New Toolbar dialog box and click OK. Now select the Button Groups tab and drag the desired buttons onto the new toolbar.

Figure 25: Creating a New Toolbar



Window Types

Origin provides a number of window types for viewing, analyzing, and presenting your data. These windows include worksheet, matrix, Excel workbook, graph, layout page, and notes windows. The window type that is currently active in the Origin workspace determines the menu bar and toolbars that are available.

Worksheet Windows

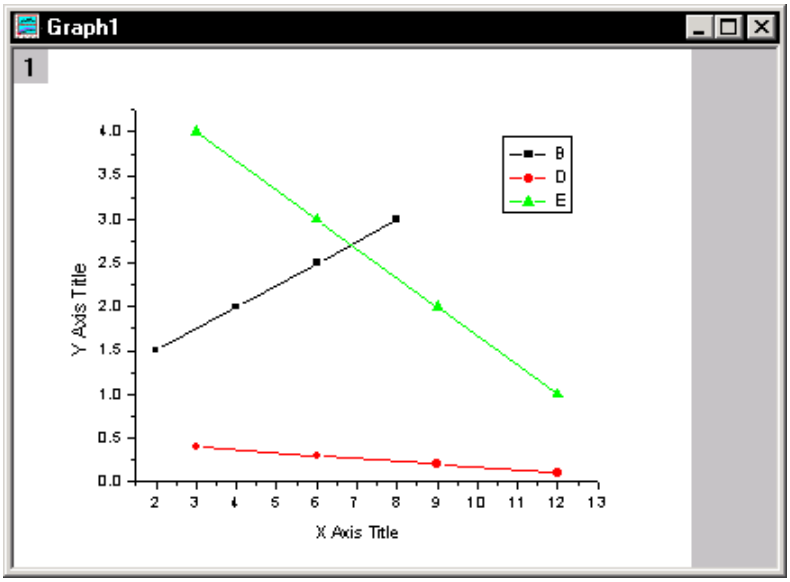
A worksheet's primary function is to hold and organize the data that you bring into Origin, and to provide tools for data manipulation, exploration, statistics, analysis, and plotting.

The columns in a worksheet are related to one another by their plotting designation (such as X and Y) and their location. The plotting designation determines how the selected data will plot. For example, in the following worksheet, if you select columns A through E and then click the Line & Symbol button  on the 2D Graphs toolbar, columns A and B, C and D, and C and E will plot as XY pairs.

Figure 26: Worksheet Column Plotting Designations

Data1

	A(X1)	B(Y1)	C(X2)	D(Y2)	E(Y2)
1	2	1.5	3	0.4	4
2	4	2	6	0.3	3
3	6	2.5	9	0.2	2
4	8	3	12	0.1	1
5					



Worksheet Menu Bar

File Edit View Plot Column Analysis Statistics Tools Format Window Help

The **File** menu provides Origin project and window commands that are available independent of the active window. When a worksheet is active, the **File** menu also provides commands to:
=> import and export data

The **Edit** menu provides cut, copy, paste, and undo commands that are available independent of the active window. Additionally, a "button edit mode" command is available for editing programmed labels. When a worksheet is active, the **Edit** menu also provides commands to:
=> set the worksheet display range
=> convert the worksheet to a matrix
=> transpose and paste transpose

Note: Undo is available for most worksheet operations, but it is not available for all Origin operations.

The **View** menu provides commands to control the display of the Project Explorer window, toolbars, status bar, and Results Log. It also provides a command to display a grid for aligning objects. These commands are available independent of the active window. When a worksheet is active, the **View** menu also provides commands to:
=> update a graph after changing the worksheet data
=> go to a specified worksheet row
=> show a hidden X column

The **Plot** menu is only available when a worksheet, matrix, or Excel workbook is active. The **Plot** menu provides commands to:
=> plot data into a specified graph template

The **Column** menu is only available when a worksheet is active. This menu provides commands to:
=> set the column's plotting designation
=> use functions to set column values or mathematically transform values
=> add and move columns

The **Analysis** menu provides commands to:
=> extract data
=> sort data

=> normalize data
=> perform FFT-related operations
=> perform nonlinear curve fitting

The **Statistics** menu is only available when a worksheet is active. This menu provides commands to:
=> perform a number of statistical tests on your data

The **Tools** menu command provides commands to set global preferences in the Options dialog box, re-activate reminder messages, and pack selected files into a single file to share with other Origin users. These commands are available independent of the active window. When a worksheet is active, the **Tools** menu also provides commands to:
=> associate LabTalk script with the worksheet
=> perform linear, polynomial, and sigmoidal fitting
=> compare two columns of data by fitting the same function to the data

The **Format** menu provides commands to display a reduced menu structure in which only basic operations are available, to edit the name and programming-related properties of a label, and to align objects to a grid. These commands are available independent of the active window.
Additionally, the worksheet **Format** menu provides commands to:
=> set the worksheet or column display options
=> set incremental X values in a worksheet without a designated X column

The **Window** menu provides commands to arrange the windows in the workspace, open the Script window, activate a Project Explorer folder, and activate a window in the workspace. These commands are available independent of the active window.

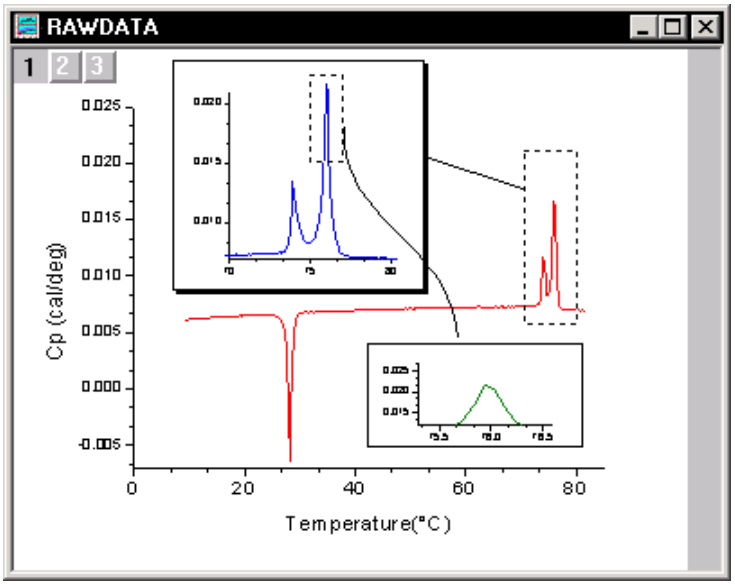
The **Help** menu provides commands to open the Origin and programming-related Help files, open a dialog box with tips on using Origin, go to the OriginLab web site, and register your copy of Origin. These commands are available independent of the active window.

Graph Windows

A graph window is a container and editor for creating graphs. Each graph window contains a single editable page. The page serves as a backdrop for the various graph objects, including layers, axes, annotations, and data plots.

There are many ways to create graphs in Origin. The easiest method is to select the worksheet data that you want to plot, and then select the desired graph type from the **Plot** menu or from one of the plotting toolbars. If you've already created a graph window, you can select and drag data from a worksheet into the graph window.

Figure 27: A Graph with Multiple Layers



Graph Menu Bar

File Edit View Graph Data Analysis Tools Format Window Help

(See the Worksheet Menu Bar for commands that are available for all window types.)

The **File** menu provides graph-specific commands to:
=> import ASCII data
=> import an image file
=> export the graph to an image file

Note: When copying or when exporting to a vector image file, you can control the graph size from the Options dialog box (**Tools:Options**, Page tab).

The **Edit** menu provides graph-specific commands to:
=> copy a graph to the clipboard
=> add and arrange layers in the active graph
=> rotate the graph between landscape and portrait orientation
=> merge all the graph windows into one window

The **View** menu provides graph-specific commands to:
=> view the graph using the screen driver or active printer driver
=> zoom in and out of the graph
=> show or hide elements in the graph
=> resize the active graph layer so that it fills the page
=> resize the graph window so that it fills the screen

The **Graph** menu provides commands to:
=> add data to the graph, including error bars or a function
=> rescale the axes and exchange the X and Y axes
=> create a new legend or color scale for color mapped data
=> stack grouped data (most useful for column and bar charts)

Note: You can also drag data from a worksheet or Excel workbook into a graph

The **Data** menu provides graph-specific commands to:
=> set the display range
=> move and remove data points
The **Data** menu also lists the data sets that are plotted in the active graph. The active data set is checked.

Note: You can open the Plot Details dialog box for a data set by pressing CTRL and selecting it from the list.

The **Analysis** menu provides graph-specific commands to:
=> perform linear and nonlinear fitting
=> perform math operations
=> smooth or filter the data
=> differentiate and integrate
=> perform a subtraction or translation
=> average curves
=> perform interpolation and extrapolation
=> perform an FFT

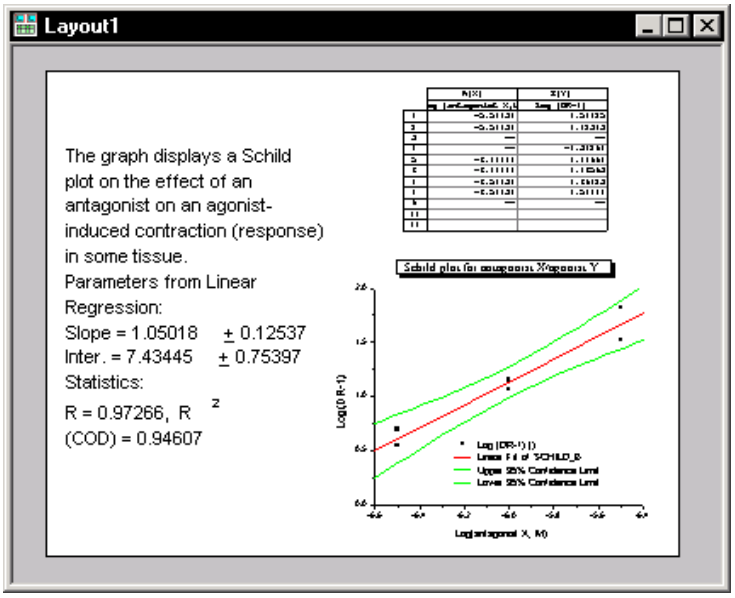
The **Tools** menu provides graph-specific commands to:
=> open analysis and fitting tools
=> open a tool for adding and arranging layers
=> compare two sets of data by fitting the same function to the data

The **Format** menu provides graph-specific commands to:
=> set the page, layer, data plot, and axes display options

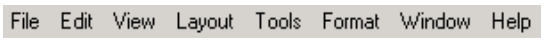
Layout Page Windows

A layout page window is a "display panel" for graphs and worksheets that have been created in other windows. You can add and arrange worksheet and graph pictures in a layout page, as well as text and other annotations.

Figure 28: A Layout Page Window



Layout Page Menu Bar



(See the Worksheet Menu Bar for commands that are available for all window types.)

The **File** menu provides layout page-specific commands to:
=> import an image file
=> export the layout page to an image file

Note: When copying or when exporting to a vector image file, you can control the size from the Options dialog box (**Tools:Options**, Page tab).

The **Edit** menu provides layout page-specific commands to:
=> copy a layout page to the clipboard
=> rotate the layout page between landscape and portrait orientation
The **View** menu provides layout page-specific commands to:
=> view the layout page using the screen driver or active printer driver
=> zoom in and out of the layout page
=> show master page items in the layout page

The **Layout** menu is only available when a layout page is active. This menu provides commands to:
=> add pictures of graphs or worksheets
=> view picture placeholders to increase redraw speed
=> view pictures using the screen driver to increase redraw speed

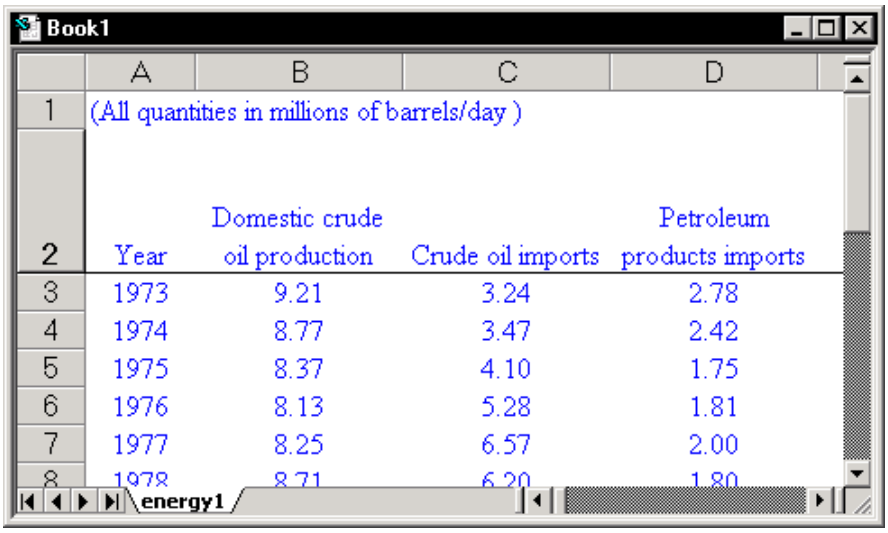
The **Format** menu provides layout page-specific commands to:
=> set the page display options

Excel Workbook Windows

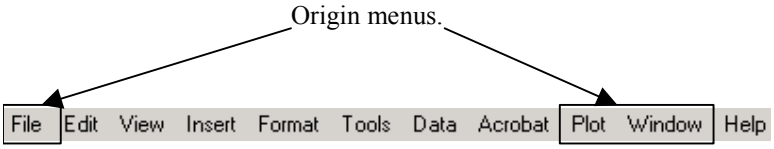
You can open Excel workbooks inside Origin, combining Origin's plotting and analysis power with Excel's spreadsheet tools. To plot your workbook data in Origin, you must have Excel version 7 (Microsoft Office 95) or later installed on your computer either as a local or network copy.

Note: When you change the active window from an Excel workbook to any other window type, the toolbar region displays a blank area where the Excel toolbars were located. To hide this spacer, right-click in this region and select **Hide Toolbar Spacer** from the shortcut menu.

Figure 29: Opening an Excel Workbook in Origin



Excel Workbook Menu Bar



When an Excel workbook is active in Origin, the menu bar displays both Origin and Excel menus. The **File**, **Plot**, and **Window** menus are Origin menus. All other menus are Excel menus.

The **File** menu provides Origin project and window commands.

The **Plot** menu provides commands to plot data into a specified graph template.

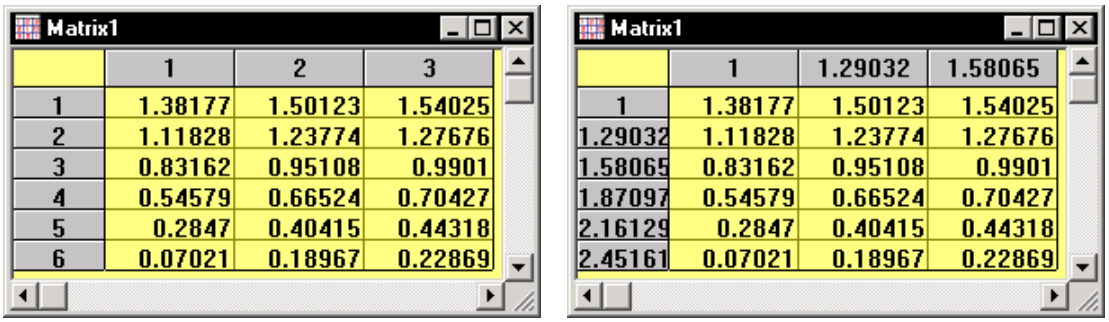
The **Window** menu provides commands to:
=> set global preferences in the Options dialog box
=> control the display of the Project Explorer window, toolbars, and Results Log
=> convert an Excel workbook to a matrix using direct conversion
=> arrange the windows in the workspace, open the Script window, activate a Project Explorer folder, and activate a window in the workspace

Note: To open the Origin Help file, right-click on the workbook title bar and select **Help Contents** from the shortcut menu.

Matrix Windows

A matrix displays a single data set containing Z values. Instead of displaying the data set as a column in a worksheet, a matrix displays the data in a specified dimension of rows and columns. A matrix is linearly mapped in X by columns and linearly mapped in Y by rows. By default, the column and row numbers display in the column and row headings. To display the X and Y values, select **View:Show XY**.

Figure 30: Showing the X and Y Matrix Values



Matrix1

	1	2	3
1	1.38177	1.50123	1.54025
2	1.11828	1.23774	1.27676
3	0.83162	0.95108	0.9901
4	0.54579	0.66524	0.70427
5	0.2847	0.40415	0.44318
6	0.07021	0.18967	0.22869

Matrix1

	1	1.29032	1.58065
1	1.38177	1.50123	1.54025
1.29032	1.11828	1.23774	1.27676
1.58065	0.83162	0.95108	0.9901
1.87097	0.54579	0.66524	0.70427
2.16129	0.2847	0.40415	0.44318
2.45161	0.07021	0.18967	0.22869

Origin provides a number of methods for converting a worksheet to a matrix. This includes direct conversion, 2D binning, converting regular XYZ data, and converting random XYZ data using gridding. These conversion methods are available from the worksheet's **Edit:Convert to Matrix** submenu.

When a matrix is active, you can create 3D, surface, contour, and image plots.

Matrix Menu Bar

File Edit View Layout Tools Format Window Help

(See the Worksheet Menu Bar for commands that are available for all window types.)

The **File** menu provides matrix-specific commands to:
=> import and export data
=> import and export image files

The **Edit** menu provides matrix-specific commands to:
=> convert the matrix to a worksheet

The **View** menu provides matrix-specific commands to:
=> when viewing an image, zoom in and out of the image
=> when viewing data, go to a specified row
=> switch between data mode and image mode
=> show the column and row numbers or show the XY mapping for the Z values

The **Plot** menu provides commands to plot data into a specified graph template.

The **Matrix** menu is only available when a matrix is active. This menu provides commands to:
=> set matrix properties such as the internal data type and the display format
=> set the matrix dimensions and the XY mapping relationship
=> use functions to set the matrix values or mathematically transform values
=> transpose, invert, shrink using averaging, expand using bilinear interpolation, smooth, or integrate the matrix

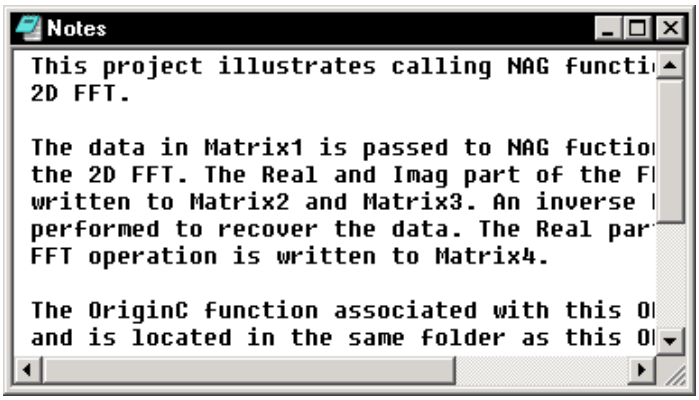
The **Image** menu is only available when a matrix is active. This menu provides commands to:
=> display images using selected color palettes
=> open a tool to adjust the brightness and contrast in the image

The **Tools** menu provides matrix-specific commands to:
=> display the Tools toolbar tools in the "region of interest mode" for selecting a region to copy, crop, or create a new matrix

Notes Windows

Notes windows can contain only text, and are thus used for recording information.

Figure 31: A Notes Window



Notes Menu Bar

(See the Worksheet Menu Bar for commands that are available for all window types.)

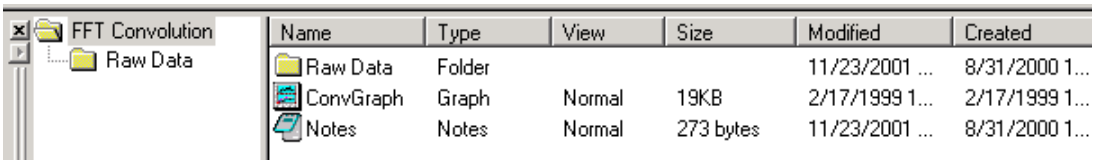
The **View** menu provides notes-specific commands to:
=> enable word wrapping

Project Explorer

Project Explorer is a tool to help you organize your Origin projects. It is particularly useful if you are working with a project that contains more than a few windows. You can use Project Explorer to develop a folder structure for organizing the windows in your project. You can also use Project Explorer to control the view of your workspace.

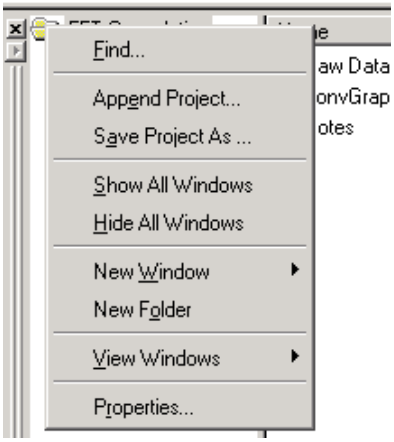
When you first start Origin, Project Explorer displays docked to the bottom edge of the workspace. You can dock it to any other edge or display it as a window in the workspace. To prevent Project Explorer from docking when positioning it as a window, press CTRL while dragging.

Figure 32: Project Explorer



To create a new folder, right-click on the project folder (or a subfolder) and select **New Folder** from the shortcut menu. Once you have created one or more subfolders, you can move windows between folders.

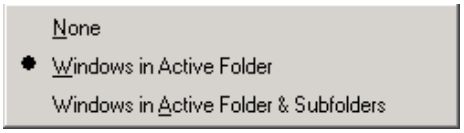
Figure 33: Creating a New Folder



In addition to adding and moving folders, Project Explorer also controls the view of your workspace. By default, only the windows in the active Project Explorer folder display in your workspace. For projects with lots of windows, this allows you to focus on specific windows in a clutter-free workspace.

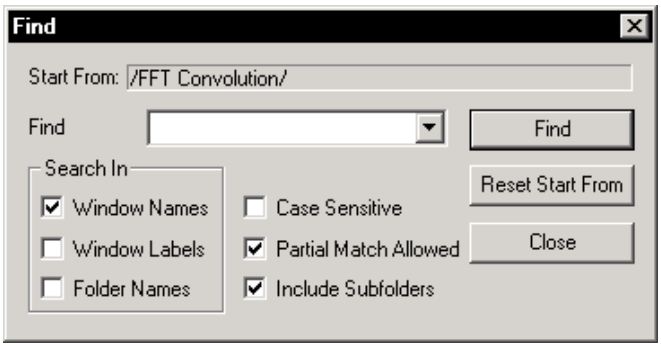
To control the view of the windows in your workspace, right-click on a Project Explorer folder or right-click in a blank space on the left side of Project Explorer. Select **View:*View Mode*** to alter the view mode. (Alternatively, select **View:View Windows:*View Mode*** from the Origin menu bar.)

Figure 34: Controlling the View of Windows in the Workspace



After you organize your windows into multiple folders, Project Explorer's Find dialog box helps you locate windows in your project. To open this dialog box, right-click on the Project Explorer folder that you want to start your search from and select **Find** from the shortcut menu. If you are unsure of the window location within your Project Explorer folder structure, right-click on the main project folder and select **Find**.

Figure 35: Finding Windows in Your Project

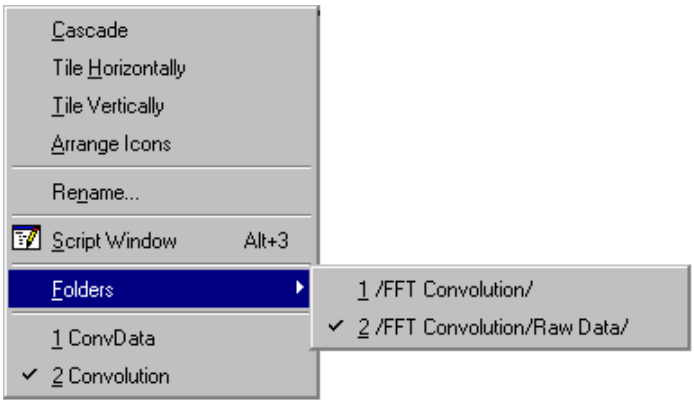


Because Project Explorer uses some of your workspace, you may want to close Project Explorer, even if you have already created a folder structure. To close Project Explorer, click the Project Explorer button  on the Standard toolbar.

When Project Explorer is closed, Origin continues to use your folder structure and view settings to display the windows in your workspace. Thus, if you are working with a project that contains subfolders, and the view mode is set to view only the active folder's windows, then Origin will display the windows in your currently active folder.

To access windows in other folders, you can re-open Project Explorer or you can select **Window:Folders:*Folder Name*** to view the windows in the selected Project Explorer folder. The window list at the bottom of the **Window** menu updates to reflect the windows in the ***Folder Name*** folder.

Figure 36: Accessing Folders and Windows from the Menu Bar



Results Log

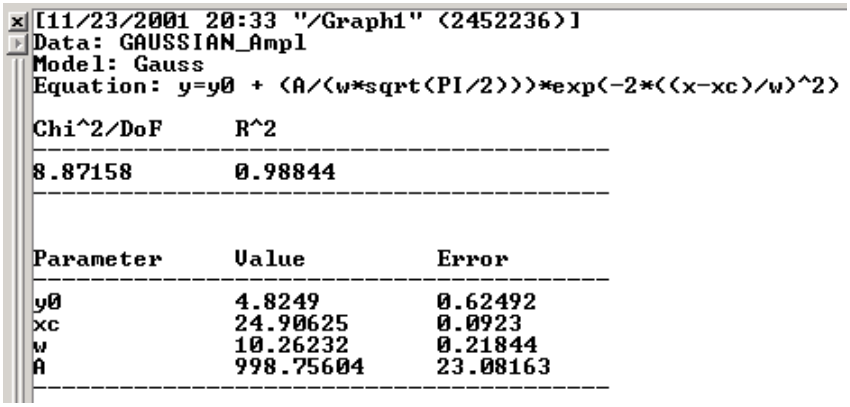
Origin automatically routes most analysis and fitting results to the Results Log. In most cases, when results are output to the Results Log, it opens automatically. However, to manually open (and close) the Results Log, click the Results Log button  on the Standard toolbar. Opening and closing the Results Log only controls its view state. You do not lose results by closing the log.

Note: When you save an Origin project, the contents of the Results Log is saved with the project

When the Results Log first opens, it displays docked to the top edge of the workspace. You can dock it to any other edge or display it as a window in the workspace. To prevent the Results Log from docking when positioning it as a window, press CTRL while dragging.

Each entry in the Results Log includes a date/time stamp, the window name, a numeric stamp which is the Julian day, the type of analysis performed, and the results.

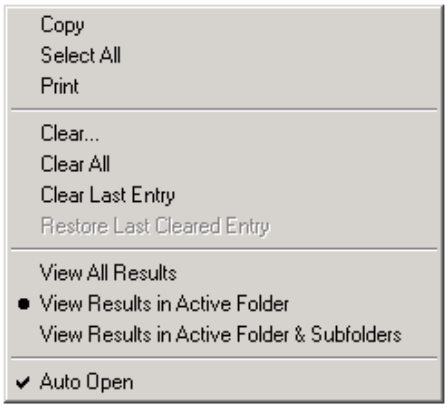
Figure 37: The Results Log



```
[11/23/2001 20:33 "/Graph1" (2452236)]
Data: GAUSSIAN_Ampl
Model: Gauss
Equation: y=y0 + (A/(w*sqrt(PI/2)))*exp(-2*((x-xc)/w)^2)

Chi^2/DoF      R^2
-----------------------------------------
8.87158        0.98844
-----------------------------------------


Parameter      Value          Error
-----------------------------------------
y0             4.8249         0.62492
xc             24.90625       0.0923
w              10.26232       0.21844
A              998.75604      23.08163
-----------------------------------------
```

You can right-click in the Results Log to open a shortcut menu with commands to copy, print, clear, and view additional results.

Figure 38: The Results Log Shortcut Menu



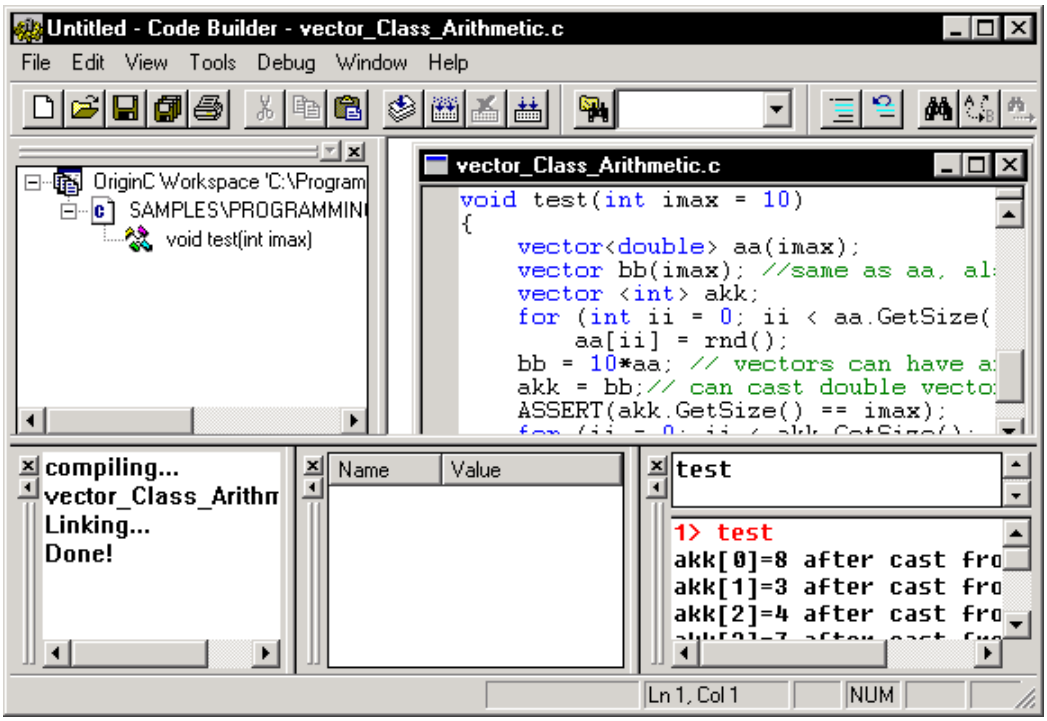
Note: When you close the Results Log after it has been docked to the edge of the workspace, the toolbar region may display a blank area where the log was located. To hide this spacer, right-click in this region and select **Hide Toolbar Spacer** from the shortcut menu.

Code Builder

Origin's programming language is called Origin C. Origin C supports a nearly complete ANSI C language syntax as well as a subset of C++ features including internal and DLL-extended classes. In addition, Origin C is "Origin aware". This means that Origin objects such as worksheets and graphs are mapped to classes in Origin C, allowing direct manipulation of these objects and their properties from Origin C.

Origin C's integrated development environment is called Code Builder. To open Code Builder, click the Code Builder button  on the Standard toolbar. Code Builder provides tools for writing, compiling, and debugging your Origin C functions. Once an Origin C function is compiled, the function becomes accessible from Origin.

Figure 39: Code Builder



To learn more about programming in Origin:

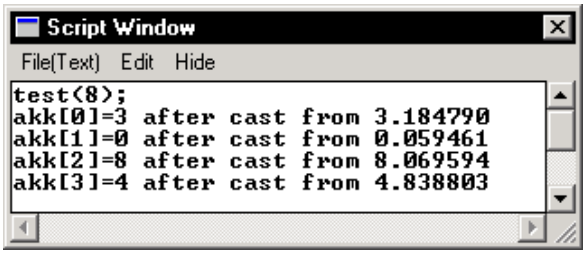
- Select **Help:Programming** from the Origin menu.
- Review the **sample Origin projects** located in the \Samples\Programming folder.
- Work through the **Command Line and Programming Tutorials** in this document.

Script Window

The Script Window is available for executing LabTalk commands. LabTalk is Origin's "historic" programming language. It has been available in versions up to and including Origin 7. However, Origin 7 introduced the new Origin C programming language. Whereas LabTalk scripts are interpreted during execution by Origin, Origin C code is compiled to byte code form and therefore executes much faster than LabTalk.

Even though Origin C is the preferred programming language in Origin 7, Origin C functions must be called using LabTalk commands. These commands can be executed from the Script window, or from many other locations that support LabTalk script. To open the Script window, click the Script Window button  on the Standard toolbar. Note that when you save an Origin project, the Script window contents are not saved with the project.

Figure 40: The Script Window



To learn more about programming in Origin, select **Help:Programming:Program Guide** or review the sample Origin projects located in the \Samples\Programming folder.

Origin Project Files

An Origin project file contains all the worksheet, matrix, graph, layout page, and notes windows that are open in the workspace when you save the project. These windows include minimized and hidden windows, as well as windows that are accessible from other Project Explorer folders in the current workspace.

Note: Origin provides sample project files in the \Samples subfolders

Excel workbook windows are a special case. If a workbook is open in the workspace, the project can be saved with a link to the source workbook file, or the workbook can be saved internal to the project.

- Saving the project with a link to the source workbook file allows you to maintain an external Excel file that is updated when changes are made to the associated workbook in Origin. This source file is also available for use in other applications and can be accessed by other users.
- Saving the workbook internal to the project enhances the project's portability. You can open the project on any computer that has Origin and Excel installed and have full access to your workbook data.
  To control how the workbook is saved with the project, right-click on the workbook window title bar and select **Properties** from the shortcut menu.

In addition to these window types, the contents of the Results Log is saved with an Origin project, as well as the current Project Explorer folder and window structure.

Saving a Project

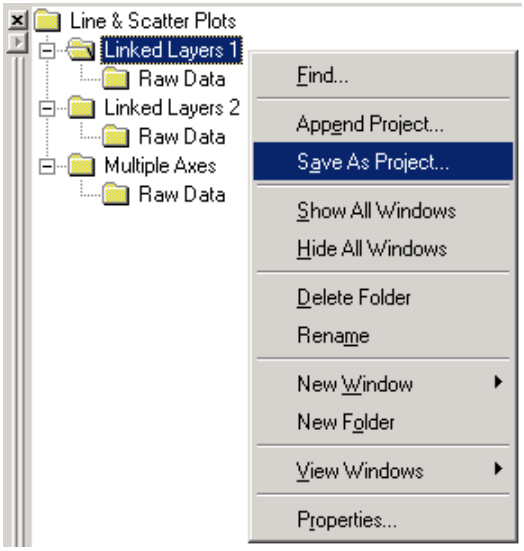
To save a project to the current file name, select **File:Save Project**. If the project has not been previously saved, this menu command opens the Save As dialog box displaying the default file name, UNTITLED.OPJ, in the File Name text box. Type the desired file name in the text box (specify the path) and click Save to save the project.

Note: For information on customizing the default save path, see "Opening an Existing Project" on page 50.

To save the project to a new file name, select **File:Save Project As**. This menu command also opens the Save As dialog box.

If you have developed a folder structure in Project Explorer, you can save a selected folder and its subfolders to a new project file. To do this, right-click on the Project Explorer folder that you want to save to a project file and select **Save As Project** from the shortcut menu.

Figure 41: Saving a Project Explorer Folder to an Origin Project File



In this example, the Linked Layers 1 folder and its Raw Data subfolder, all the windows that they contain, plus Results Log output specific to these windows are saved to a new project file.

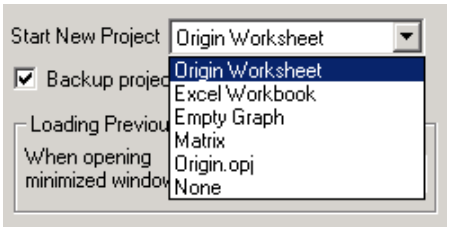
Automatically Creating a Backup

Origin will automatically create a backup of the currently saved project file before re-saving. Origin renames the saved file BACKUP.OPJ, and then saves the altered file using the specified file name. The automatic backup feature is set on the Open/Close tab of the Options dialog box (**Tools:Options**).

Opening a New Project

A new project is opened each time you start Origin. You can also open a new project by clicking the New Project button on the Standard toolbar. In both cases, the new project contains one worksheet window. You can, however, customize the new project that opens. To do this, select **Tools:Options** to open the Options dialog box. Select the Open/Close tab and then modify the Start New Project setting.

Figure 42: Customizing the New Project Setting



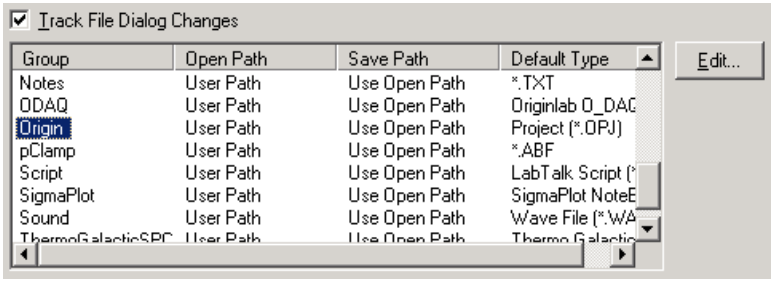
You can select from worksheet, Excel workbook, empty graph, matrix, ORIGIN.OPJ, or none. The ORIGIN.OPJ file includes a worksheet and a graph window. The graph window is configured to display a data plot of column A (X values) and column B (Y values) automatically, after you import or enter data into the worksheet columns. You can customize the ORIGIN.OPJ project file and resave it. After you click OK in the Options dialog box, you are asked if you want to save this setting for future Origin sessions.

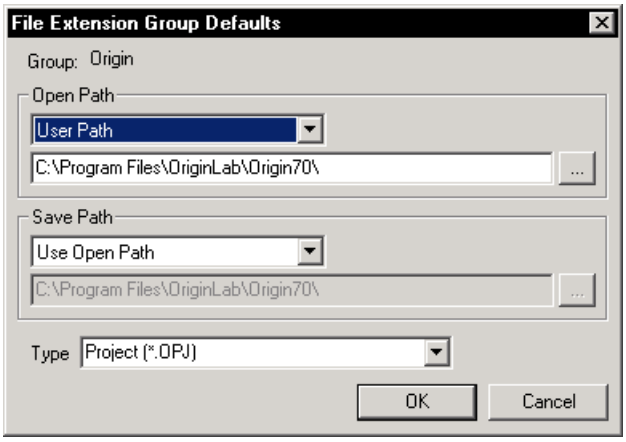
Opening an Existing Project

To open an existing project file, click the Open button  on the Standard toolbar and select the desired file from the Open dialog box.

By default, Origin will keep track of the project file path, so that each time you open a project in the current Origin session, the path to the last opened file will be selected by default in the Open dialog box. To turn off file tracking or to specify a custom project file path, select **Tools:Options** to open the Options dialog box. Select the File Locations tab. Select Origin from the top list box and then click Edit. This opens the File Extension Group Defaults dialog box.

Figure 43: Customizing File Tracking





- To specify a custom project file path, select User Path in the Open Path group and then browse to the desired folder from the associated text box. Click OK.
- To set your Origin folder as the project file path, select Origin Path from the Open Path drop-down list. Click OK.

To direct Origin to always use this custom path, clear the Track File Dialog Changes on the File Locations tab. Click OK. An Attention dialog box asks if you want to save this setting for future Origin sessions. Click Yes.

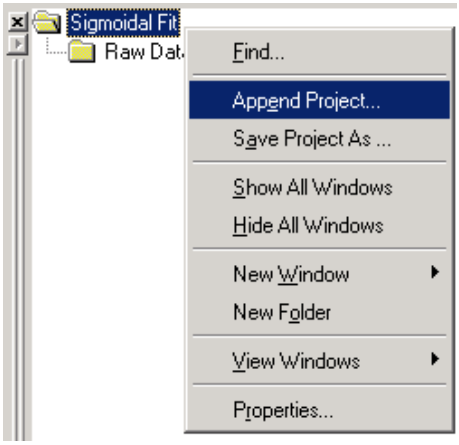
Opening More than One Project

Origin can only open one project at a time. To open multiple projects, you must start multiple instances of Origin.

Appending Projects

The contents of a project can be added to the currently opened project by selecting **File:Append** or by right-clicking on a Project Explorer folder and selecting **Append Project** from the shortcut menu.

Figure 44: Appending Projects



After you select the project to append, Origin displays a Reminder Message asking if you want to append the contents of the new project into a new Project Explorer folder or into the current folder.

Note: To re-activate the display of Reminder Message boxes, select **Tools:Reactivate Reminder Messages**.

If duplicate window names exist between projects, Origin automatically changes the name of the appended window by adding an "A" to the window name (*AWindowName*). For example, in the following figure, a project containing a Data1 worksheet is appended to a project that also contains a Data1 worksheet. The appended window is renamed AData1.

Figure 45: Renaming Duplicate Window Names when Appending

Project Windows

Project windows include worksheet, matrix, Excel workbook, graph, layout page, and notes windows. You can create and open any number of windows in an Origin project. The only limitation is your machine's memory. However, you should keep your projects manageable by limiting the number of windows contained in a project. You should also consider organizing the windows using Project Explorer.

Note: To learn how to organize your project using Project Explorer, see "Project Explorer" on page 43.

Creating a New Window

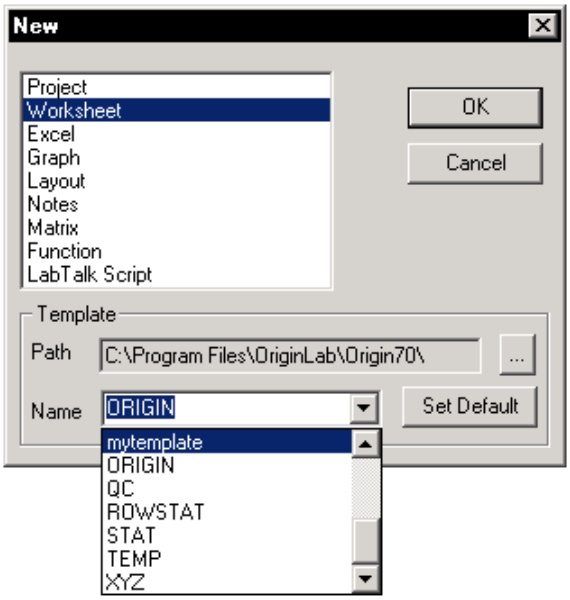
To create a new window, click one of the New *Window* buttons on the Standard toolbar.

Figure 46: New Window Buttons on the Standard Toolbar

These buttons create windows based on a default template. For example, the New Worksheet button  creates a worksheet based on the ORIGIN.OTW template. You can change the template that is associated with any of these buttons. To do this, select **File:New** to open the New dialog box. To change the template associated with the New Worksheet button, select Worksheet from the list box, then select the desired template from the Template group. Click Set Default to associate this template with the New Worksheet button.

Note: To learn more about templates, see "Window Templates" on page 55.

Figure 47: Changing the Template for the New Worksheet Button



You can also use the New dialog box to create new windows based on specific templates.

Renaming a Window

To rename a worksheet, matrix, graph, layout page, or notes window, activate the window and then select **Window:Rename**. Alternatively, right-click on the window's title bar and select **Rename** from the shortcut menu. Both menu commands open the Rename dialog box.

To rename an Excel workbook window, right-click on the window's title bar and select **Properties** from the shortcut menu.

When renaming windows, consider the following:
=> The window name cannot start with a numeral.

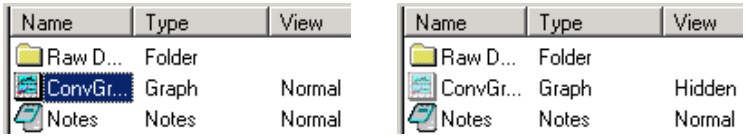
=> Origin ignores any spaces between characters.
=> The name cannot exceed 13 characters in length.

For greater flexibility in window naming, include the Label field in the window title bar. The label allows more characters and accepts special characters (for example, an underscore and dollar sign). It also allows spaces between characters.

Hiding a Window

To maximize the use of your workspace, you can hide windows from view - without deleting them from the project. To hide a window in the workspace, select the desired window icon on the right side of Project Explorer and then double-click on this icon. To return the window to the normal view mode, double-click again on the window icon. These view controls are also available by right-clicking on the window icon in Project Explorer, or by right-clicking on the window title bar.

Figure 48: Hiding a Window from View



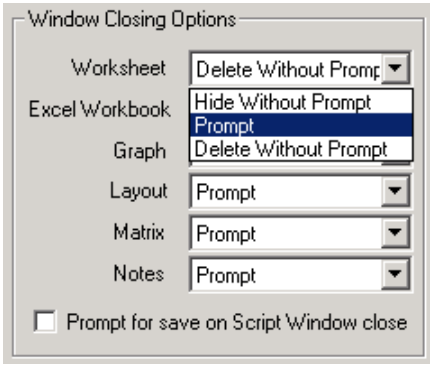
Deleting a Window

To delete a window from the project, click the X button in the upper-right corner of the window. Origin then asks you if you want to hide or delete the window.

Note: When you delete a window, it is no longer a part of the project.

To prevent this verification prompt from displaying, select **Tools:Options** to open the Options dialog box. Then select the Open/Close tab. In the Window Closing Options group, select how you want Origin to respond when you click a particular window type's close button.

Figure 49: Setting the Close Option for a Window Type



Refreshing a Window

Whenever you change the contents of a graph window or a worksheet, or expose a part of a window that was hidden, Origin automatically redraws the affected portion of the window. Occasionally, a portion of

the window may not refresh correctly. Whenever this occurs, click the Refresh button  on the Standard toolbar to refresh the active window.

Note: To control the percentage of the window that must be exposed before Origin automatically redraws the window, select **Tools:Options** to open the Options dialog box. Select the Miscellaneous tab and then edit the No Redraw (% Covered) combination box.

Duplicating a Window

To duplicate the active worksheet, graph, matrix, or layout page window, click the Duplicate button  on the Standard toolbar. When a duplicate window is created, Origin names the window using the default window name syntax:

DefaultWindowNameN

where *N* is assigned the lowest available number not used by other window names (of the same type) in the project.

Table 1: Default Window Names

Window	Default Name
Worksheet	Worksheet*N*
Graph	Graph*N*
Matrix	Matrix*N*
Layout Page	Layout*N*
Function Graph	Function*N*

Saving a Window

In addition to saving projects, windows (except for layout pages) can be saved to a file. When a window is saved to a file, the window can be opened in any other Origin project. To save the active window to a file, select **File:Save Window As**. This menu command opens the Save As dialog box. Origin automatically lists the correct file extension for the active window type in the Save as Type drop-down list.

Table 2: Window File Extensions

Window	File Extension
worksheet	.OGW
matrix	.OGM
graph	.OGG
Excel workbook	.XLS
notes	.TXT

Opening a Window from a File

If you have a window in one project that you want to include in another project, you can append the projects. However, this action will add all the windows from a selected project into the currently open project. To "append" only the desired window, you must save the window to a file and then open the

window in the desired project. To open a worksheet, graph, matrix, or notes window that was saved to a file, click the Open button on the Standard toolbar. This button opens the Open dialog box. From the Files of Type drop-down list, select:

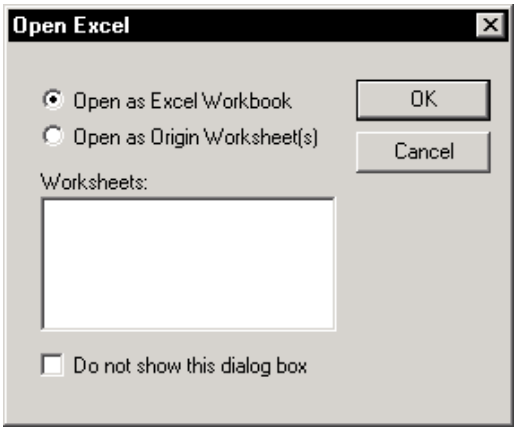
Note: To learn how to append projects, see "Appending Projects" on page 51.

'Worksheets (*.OGW)' to open a worksheet window.
'Graphs (*.OGG)' to open a graph or function graph window.
'Matrix (*.OGM)' to open a matrix window.
'Text (*.TXT)' to open a notes window.

After you open a worksheet, graph, matrix, or notes window file in a project and then save the project, the window then remains a part of this project.

To open an Excel workbook file, click the Open Excel button on the Standard toolbar. After you select your Excel file, a dialog box opens asking if you want to open the workbook as an Excel workbook or as an Origin worksheet.

Figure 50: Opening an Excel Workbook



If you open the workbook as a workbook, you can continue to use Excel's spreadsheet tools to process your data, all within the Origin workspace. You also have access to Origin's plotting and analysis tools. When you are ready to save your project, you can save the project with a link to your source workbook, and update the link, or you can save the workbook as part of your Origin project. When you update the link to a (linked) workbook, the updated workbook is available for other applications or other users.

If you open the workbook as one or more Origin worksheets, you have no access to Excel's spreadsheet tools in Origin. Furthermore, the data no longer has a connection to the source workbook. Thus, the changes you make to the data are lost to the original workbook.

Window Templates

When you create a new worksheet, graph, or matrix window, (for example, by clicking the , , or button on the Standard toolbar), Origin creates the new window based on a template. The template determines the properties of the new window. For example, if the new window is a worksheet, the template determines the number of columns in the worksheet, each column's plotting designation and display type, the ASCII import settings, and any formulas used to set the column values. If the new window is a graph, the template determines the number of layers (sets of XY axes) on the page and their arrangement, the

types of data plots in each layer (for example, scatter or bars), the number of data plots, the axes scale type, and text labels and other annotations. Basically the template determines all the attributes of the window, except for the actual data the window will contain. Data is never saved with a template.

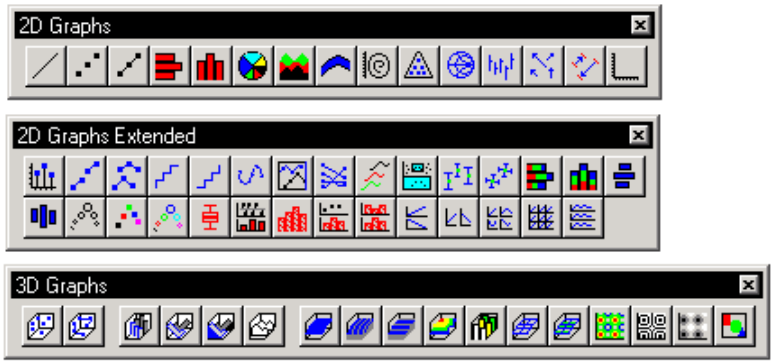
The following table lists the file extensions for the templates.

Table 3: Template File Extensions

Window	Template File Extension
worksheet	.OTW
graph or function graph	.OTP
matrix	.OTM

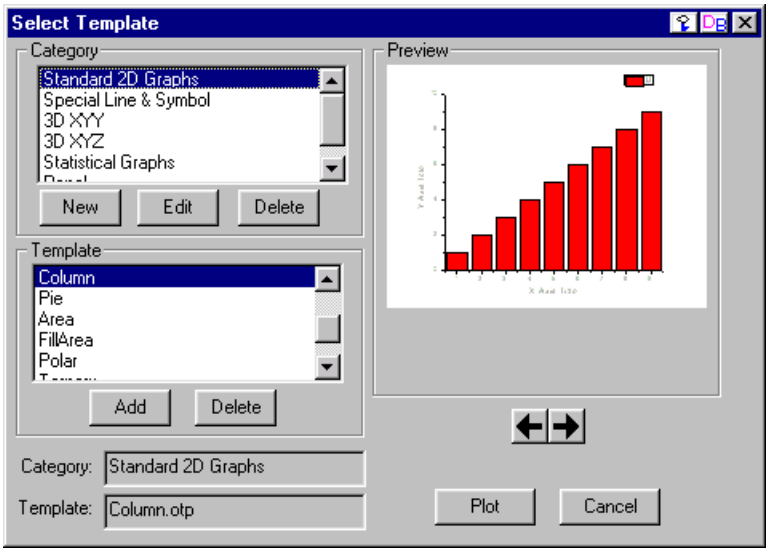
Origin provides many built-in templates. For example, most of the graph templates are accessible from the plotting toolbar buttons.

Figure 51: Plotting Toolbar Buttons



Origin also provides a Template Library tool for categorizing and accessing graph templates. To open the Template Library tool when a worksheet or an Excel workbook is active, select **Plot:Template Library**.

Figure 52: The Template Library Tool



You can also use the Template Library tool to plot your worksheet or Excel workbook data. If you highlighted data in the worksheet or workbook before opening the tool, and your data selection is appropriate for the template you've selected, then click the Plot button to plot the data into the template. If you did not highlight data or if your selection was not appropriate for the template you've selected, then click the Plot button to open an intermediary dialog box for data selection.

To create a custom template, you can modify an existing template or create a new template. The procedure is similar in both cases. You must first open a window based on a built-in template, customize the properties of the objects in the window, and then save the window as a template. The instructions on re-building the window are saved with the template. However, any data in the window is not saved with the template.

For example, suppose you want to customize the template associated with the New Worksheet button  on the Standard toolbar. To do this, click  to open a new worksheet. Now customize the attributes of this worksheet. For example, perhaps you always import ASCII data files that contain data in the format X, Y, Y Errors. In this case, click the Add New Columns button  on the Standard toolbar. Then right-click on the new column (C) and select **Set As:Y Error** from the shortcut menu.

To save this window as a template, select **File:Save Template As**. The built-in template ORIGIN.OTW is listed by default in the File Name text box. If you click Save, you will overwrite the built-in template with your custom settings. Alternatively, you can type a new file name to save your changes to a new template.

If you save to a new file name, you can associate the New Worksheet button  with this new template file (it is currently linked to ORIGIN.OTW). To do this, select **File:New** to open the New dialog box. Select Worksheet from the list box. Then select your new template file from the Template group. Finally, click the Set Default button to associate this template with the New Worksheet button.

Origin Tutorials

The Origin Tutorials focus on the functionality of Origin's user interface. Tutorials cover data import, plotting, use of some of Origin's data exploration and analysis tools, nonlinear curve fitting and the basics of creating a graphic presentation.

Tutorial 1, The Origin Workspace

This tutorial will introduce you to components and use of the Origin workspace. We will discuss Origin's windows, menus, toolbars, and the Project Explorer workspace.

Origin Windows

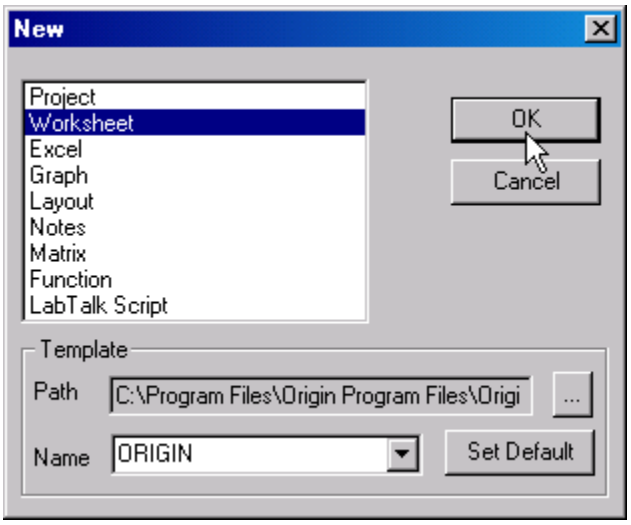
Origin has numerous windows and workspaces available for completing various tasks. The **worksheet** is the primary structure for organizing your data. On the surface, the worksheet looks much like a conventional spreadsheet. However, its behavior and capabilities are unique. You will become familiar with its use as you begin to work with Origin.

When you start Origin, the default Origin workspace opens to display a single worksheet named Data1.

1. If you have not done so, start Origin. Note the single blank worksheet window named Data1.

To open a fresh Origin worksheet:

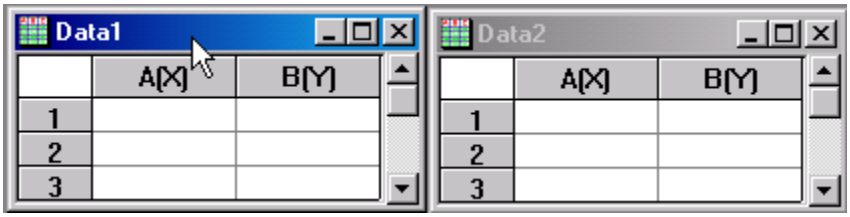
2. From the menu, choose **File:New**. The **New** dialog box opens.



3. Select **Worksheet** and click **OK**. A second worksheet is added to the workspace. Note that Origin has automatically named the worksheet Data2.

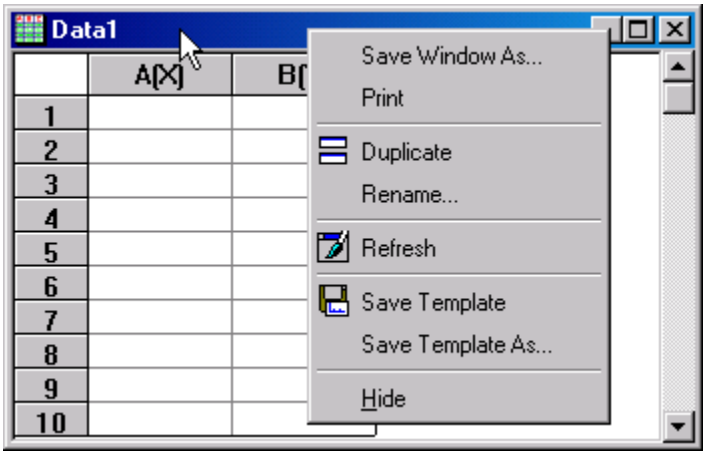
Notice also that as you opened the new worksheet, the color of the title bar of the Data1 worksheet turned gray. The new worksheet's title bar is now colored, indicating that it is the *active window* in the Origin workspace. Any commands that you issue to Origin, either through the user interface (the menus, toolbars, etc.) or programmatically (via LabTalk or Origin C), act upon the active window. It is likely that many windows of many different types will be open in the Origin workspace, and the color of the title bar is your primary indicator as to which is the active window.

4. Click once anywhere on the Data1 worksheet. The title bar now becomes colored and Data1 becomes the active window. The title bar of Data2 is now gray.



Origin keeps track of the various windows in the workspace by assigning a unique name to them. You can assign your own names to worksheet windows, but Origin will not allow you to assign the same name to two windows. Recall that when you opened a second worksheet, Origin automatically assigned the name of Data2 to the worksheet to distinguish it from Data1. The necessity for this naming system will become more apparent as you work with Origin, but for now, suffice it to say that the window name is a unique identifier that allows for manipulation of the window and its contents.

To rename a window:



5. Right-click once on the window title bar of the Data1 worksheet and select **Rename** from the shortcut menu. The **Rename** dialog box opens.
6. Enter **MyData1** in the Name field but *do not* click OK just yet.

In the Label field enter **Data from Trial 1**, select the **Both Name and Label** radio button, and click **OK**. Note that spaces are not allowed in window names but spaces are okay in window labels. Learn more about window names and labels.

7. Click the **Close** button in the upper right corner of the Data2 worksheet. An attention box opens asking you to delete or hide the worksheet. Choose **Delete**. The worksheet closes.
8. Click the **Close** button in the upper right corner of the renamed MyData1 worksheet. An attention box opens asking you to delete or hide the worksheet. Choose **Delete**. The worksheet closes.

To learn more about Origin worksheets, *see* **Tutorial 2, Worksheets, Data Import and Plotting**.

If you are an Excel user, you were probably quite interested to hear that Origin not only opens Excel files, but allows you to create, edit, analyze, graph, and save Excel workbook data without leaving the Origin workspace. We will look further at this in a later tutorial, but, for now, let's just open a fresh Excel workbook.

To open an Excel workbook window:

9. From the Origin menu, choose **File:New**. The **New** dialog box opens.

10. Select **Excel** and click **OK**. A blank workbook opens. Note that the workbook is named Book1.

Note: It may take a moment to open the Excel workbook. In fact, if you are observant, you'll notice that the program menu structure changes completely as the workbook opens. This is because you are actually running the Excel program within the Origin workspace. Thus, the menus that you see when the workbook opens are actually Excel's menus.

The Origin graph window is the primary window for plotting in Origin. Origin's many built-in 2D and 3D graph types use very different specifications (and graph templates) to create a graph, but all graphs – whether 2D or 3D – use the same Origin graph window.

To open an Origin graph window:

11. Choose **File: New** from the program's main menu. The **New** dialog box opens.
12. Select **Graph** and click **OK**. A blank graph window opens. Note that the graph window is named Graph1.

Most often you will use the worksheet for tabulating and manipulating your data, while you will use the graph window for plotting your data. However, if you are making 3D surface or contour plots of XYZ data, you will need to become familiar with another window type – the Origin matrix.

The Origin matrix, like the worksheet, superficially resembles a spreadsheet, though it's purpose is to store the Z coordinates of an XYZ dataset. We will say much more about Origin matrices in a later tutorial, but, for now, let's just have a look at one.

To open an Origin matrix window:

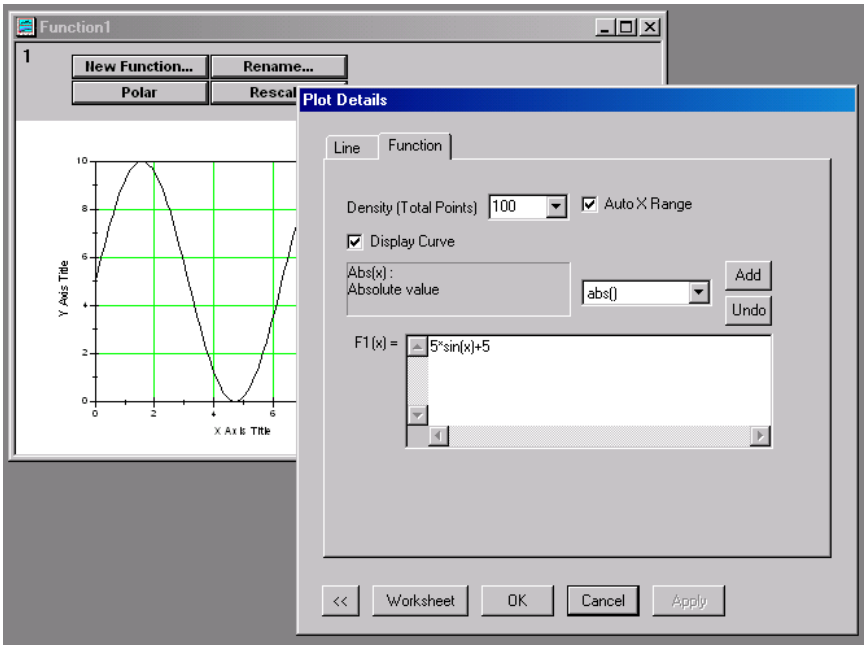
13. From the menu, choose **File:New**.
14. Select **Matrix** and click **OK**. A data window with a pale yellow background opens. The window is named Matrix1.

The same rules that govern the naming of worksheets and graph windows apply to matrix windows.

Origin has another special type of graphing window called a Function window. This window allows you to plot a function without having to create a dataset. Consult you Origin Help file (**Help:Origin**) for more information on the Function graph.

To open an Origin Function window:

15. From the menu, choose **File: New**.
16. Select **Function** and click **OK**. A blank Function window opens. Simultaneously, the **Plot Details** dialog box will open to the **Function** tab.



Note: The **Plot Details** dialog box is Origin's all-in-one interface for customizing Origin graphs. You will learn more about the Plot Details dialog in later tutorials.

Of course, much of Origin's utility derives from its presentation graphics capabilities and Origin's **Layout** window is specifically designed for assembling graphic presentations. While you *can* combine several graphs on a single graph page, the Layout window is the *only window in which you can combine worksheets and graphs*.

To open an Origin Layout window:

17. From the menu, choose **File:New**.
18. Select **Layout** and click **OK**. A blank Layout window opens.
19. Right-click in the gray area to the right side of the Layout page (this is the area outside the page but *inside* the window).
20. Choose **Rotate Page** from the shortcut menu. The page orientation changes from landscape to portrait.

The **Notes** window is a general purpose text window that is useful for making notes to yourself about procedures, to pass information on to others with whom you share your Origin files, to save equations, scripts, mathematical functions, etc.

To open an Origin Notes window:

21. From the menu, choose **File: New**.
22. Select **Notes** and click **OK**. A blank Notes window opens.

Origin has another specialized text window that reports and stores your analytical output (statistical results, curve fitting results, etc.). This is the **Results Log**.

To display the Results Log:

23. From the menu, choose **View:Results**.

This menu item toggles the Results Log on and off, so that you can conserve screen space when it is not needed. Hiding the Log does not delete the contents; unless cleared, Results Log contents are saved with the Origin Project file.

Another cornerstone of Origin's power is its programmability. Origin has two programming languages: LabTalk and Origin C. **LabTalk** is Origin's proprietary scripting language. The Script window provides a convenient way to issue LabTalk commands in Origin.

To display the Script window:

24. From the menu, choose **Window:Script Window**. A blank Script window opens.
25. Close the Script window by clicking the **Close** button in the upper right corner of the window.

Origin C is Origin's new ANSI C-compliant programming language. Origin provides a special workspace known to programmers as an integrated development environment (IDE). This Origin IDE, called **Code Builder**, is provided for writing, compiling, running and debugging your Origin C programs.

To display the Code Builder integrated development environment:

26. Choose **View:Code Builder** from the menu. The Code Builder IDE opens.
27. Click the **Close** button in the upper right-hand corner of Code Builder.

There can only be one instance of Code Builder or one Script window opened per instance of Origin. You can toggle the Script window on and off without losing the window contents. To temporarily hide Code Builder, use the minimize button in the upper right hand corner of Code Builder. To learn more about programming in Origin, see the Command Line and Programming Tutorials.

Origin Menus

Our quick tour of Origin's many window types probably gave you a hint as to the wide range of Origin's capabilities. In order to accommodate Origin's varied capabilities, Origin's designers have created a **context sensitive** menu structure. Context sensitive menus present the user with only the commands that are pertinent to the currently active window type (worksheet, graph, matrix, etc.). This helps to speed your work, because you are not forced to wade through dozens of irrelevant or grayed-out menu items to find the one that you are looking for.

To see how this context sensitive menu structure works, do the following:

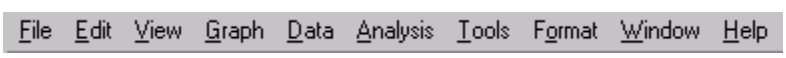
1. If you have not done so, start Origin. If you are continuing from the previous Origin windows tutorial, select **File:New Project**. When prompted to save the contents of your previous session, say **No**.

At this point, you should have a clean workspace containing a single Origin worksheet.

Origin's main menu should look like this:

2. From the menu, choose **File:New**. The **New** dialog box opens. Select **Graph** and click **OK**. A blank graph window opens.

Origin's main menu should now look like this:



Note that the menu options have changed. Opening a new graph window made the graph window the active window, and Origin altered the menu structure to *hide worksheet-relevant* options and make *graph-relevant* options available to you.

3. From the menu, choose **File:New**. The **New** dialog box opens. Select **Matrix** and click **OK**. A blank matrix window opens.

Origin's main menu should now look like this:

Again, note that the menu structure has changed. When the new matrix window became the active window, the menu structure changed to make matrix-relevant commands available to you. Let's perform one more operation just to reinforce the point of this short exercise: that Origin's menu structure changes according to the active window type. If you find that you are suddenly unable to find the menu command that you are looking for, chances are good that you simply need to activate a window of the appropriate type.

4. From the menu, choose **File:New**. The **New** dialog box opens. Select **Excel** and click **OK**. A blank Excel workbook window opens.

Origin's main menu should now look like this:

Note: it may take a moment to open the Excel workbook. When you choose to open an Excel workbook in Origin, you are actually running the Excel program within the Origin workspace. Thus, there may be a several second delay before the program responds.

Origin Toolbars

Origin's toolbars give you quick access to many of Origin's menu commands. This tutorial will give you a few pointers on managing your Origin toolbars.

As you work with Origin, you may find that there are some toolbars that you use extensively and others that you use rarely, if at all. You can choose to display or hide any built-in or user-defined toolbar.

To hide or display an Origin toolbar:

1. Select **View:Toolbars** from the menu. The **Customize Toolbar** dialog opens.

The **Toolbars** tab displays a list of standard Origin toolbars. Those that are active (visible in the workspace) will have a check mark next to them.

2. Position your cursor over the check box next to the **Standard** toolbar and click once to clear the box. Notice that the Standard toolbar disappears from the Toolbar spacer.
3. Position your cursor over the check box next to the Standard toolbar and click once to select (restore) the Standard toolbar.

In addition to hiding or displaying toolbars, you can use the Customize Toolbar dialog to add or remove buttons from your toolbars or create new toolbars using a single button or a combination of buttons.

We should point out that when you choose **View:Toolbars** from the menu, you enable toolbar edit mode. To see how this works, let's run through a short exercise.

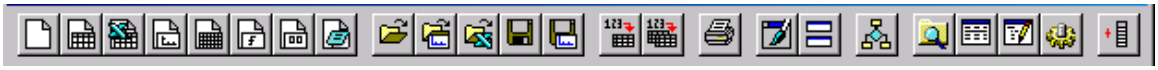
To customize a toolbar:

4. Place your cursor over the **New Project** button  on the **Standard** toolbar (if you closed the Customize Toolbar dialog, reopen it.).

5. Hold down the left mouse button, drag the **New Project** button into a an empty region of the Origin workspace and release the mouse button. Notice that the New Project button is removed from the Standard Toolbar.

To add a button to an Origin toolbar:

6. Click on the **Button Groups** tab of the Customize Toolbar dialog.
7. Select **Standard** in the toolbar **Groups** list. Note that the Standard toolbar buttons appear to the right under the **Buttons** heading.
8. Place your cursor over the **New Project** button  and drag the button icon back to the first position on the Standard toolbar. The complete Standard toolbar looks like this:



As you become more familiar with Origin, you may decide to create your own toolbars using just the buttons that you use most, or you may wish to take it a step further and create your own bitmaps and buttons that perform custom routines. To find out more about Origin toolbars, select **Help:Origin** and search on keyword **toolbars**.

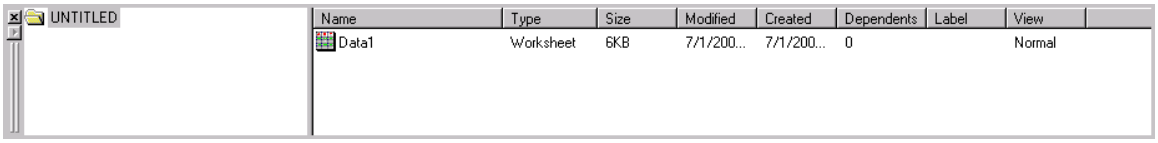
Project Explorer

Typically, users amass quite a lot of data in an Origin project file. If you anticipate building a project file that contains dozens of worksheets, graphs, notes windows, etc., you will probably want to use **Project Explorer** to help you manage your Origin workspace. Project Explorer helps to organize your workspace so that you see only data that is relevant to the task at hand. In addition, you can use Project Explorer to create new project files from a portion of an existing project file or to append the contents of another project file to your current file. The Project Explorer workspace can be hidden or restored as needed.

To display the Project Explorer workspace:

1. If you are continuing from a previous session, click the **New Project** button  on the Standard toolbar. You do not need to save anything from a previous tutorial session.
2. If the Project Explorer workspace is not open, press **ALT + 1**, or click on the **Project Explorer** toolbar button  .

The Origin workspace should only have a single worksheet named Data1. The Project Explorer window will display at the bottom of the Origin workspace and should look something like this:



The left hand window shows a single folder entitled “Untitled.” This is your as-yet-unnamed Origin project file. The right hand window displays the contents of your project file – at this point, a single worksheet entitled Data1.

3. Add another Origin worksheet by clicking on the **New Worksheet** button  on the Standard toolbar.

Note: The New Worksheet button performs the same operation as selecting **File:New…Worksheet** from the menu.

Your unnamed project file should now contain two worksheets and both should display as icons in the Project Explorer workspace.

4. Click the **New Graph** button  on the Standard toolbar.
5. Click the **New Graph** button once more. You should now have two worksheets and two graphs in the untitled project file.

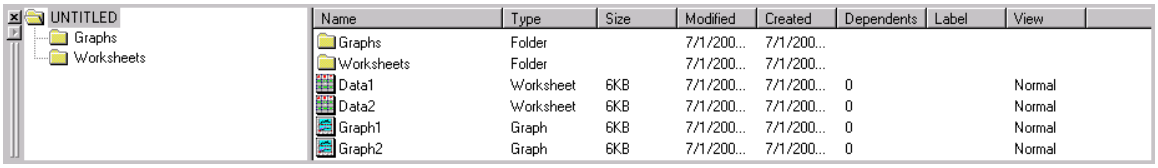
Obviously, adding many new worksheets and graphs to the project would, in time, make it difficult to find what you are looking for. To help manage your project files, you might create subfolders within the Origin project file. Let's create two subfolders – one for worksheets and one for graphs – and put the empty worksheets and graphs that we have opened into the appropriate folder.

To add a subfolder to the project using the Project Explorer interface.

6. Right-click on the Untitled folder icon in the left-hand pane of the Project Explorer workspace. A shortcut menu opens.
7. Select **New Folder** from the shortcut menu.

A New Folder opens in the right-hand pane of Project Explorer. You may rename this folder at this time. We'll name it **Worksheets**.

8. Once again, right-click on the Untitled folder icon in the left-hand pane of the Project Explorer workspace.
9. Again, select **New Folder** from the shortcut menu and rename it **Graphs**.



Your Project Explorer workspace should now look something like this.

10. Hold down the **CTRL** key and select Data1 and Data2 from the right-hand pane. When you have selected both of them, hold down your left mouse button and drag the two window icons to the **Worksheets** subfolder on the left side of the workspace.
11. Hold down the **CTRL** key, select Graph1 and Graph2, and drag the two graph window icons to the **Graphs** subfolder.

Your worksheets and graphs are now in two separate subfolders. How you organize your Origin project files is, of course, entirely up to you, and will depend on your needs and your work style. Suppose that we wanted to save just the contents of the Graphs subfolder as a separate file?

To save a project subfolder as a separate file:

12. Click on the **Graphs** subfolder in the left-hand pane, right-click and choose **Save Project As…**from the shortcut menu. Origin's **Save As** dialog opens prompting you for a name and destination directory. You do not need to save this file.
13. Choose **Cancel**.

On occasion, you may also find it helpful to hide certain windows, so that they are not visible in the Origin workspace These may be worksheets of questionable data or the results of something that you tried that didn't work out. Rather than delete these windows, you may prefer to simply hide them.

To hide or show a window in Origin using the Project Explorer:

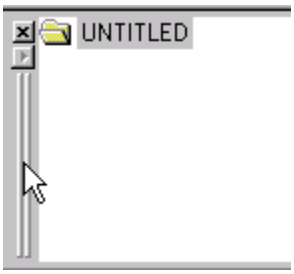
14. Select the **Graphs** subfolder in the left-hand pane of the Project Explorer workspace. Your two graph windows should be visible in the Origin workspace. Notice also, that two window icons – Graph1 and Graph2 – appear in the right hand pane of Project Explorer.
15. Click the **Close** button in the upper right hand corner of Graph1. An attention box asks whether you wish to **hide or delete** Graph1.
16. Choose **Hide**. Notice that Graph2 disappears from the Origin workspace. Notice also that, although the Graph2 icon is still visible in the right-hand pane of Project Explorer, it is a bit "washed out" looking. This is a visual cue as to the **hidden** status of this window.
17. Double-click on the Graph1 icon in the right hand side of Project Explorer. The graph window is restored and the window icon now looks "normal."
18. Double-click again on the Graph1 window icon in the right-hand pane of Project Explorer. The window is again hidden.

Thus, you see that there are a couple of ways that you can show or hide windows in the Origin workspace.

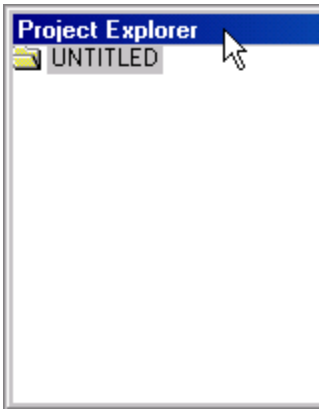
The last part of this exercise is simply a short lesson in manipulating the position of the Project Explorer workspace. The default state for Project Explorer is docked at the bottom of the Origin workspace. However, this can be altered, as needed.

To undock and dock the Project Explorer workspace:

19. To **undock** Project Explorer, double-click on the two vertical bars on the left side of the Project Explorer workspace.



20. To return the Project Explorer workspace to it's **docked** position, double-click on the workspace title bar.



In terms of default position, the Project Explorer workspace behaves as do Origin toolbars. When you exit the program, the existing position and state (docked or undocked) of toolbars or of the Project Explorer workspace are stored as the default.

This concludes the **Origin Workspace** tutorial.

Tutorial 2, Worksheets, Data Import, and Plotting

The worksheet is Origin's primary data storage structure.

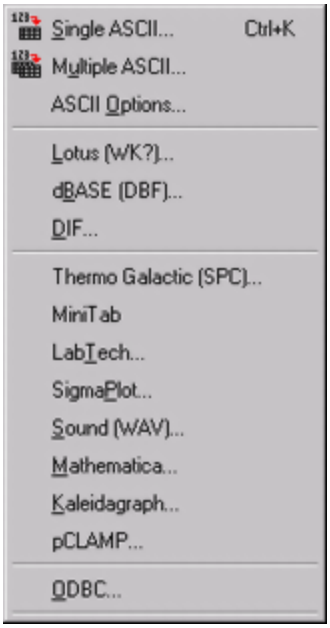
In this tutorial, we import a sample dataset into an Origin worksheet and format the worksheet to control the way that the data are plotted. Then, we create a line and symbol graph from our imported data.

The Origin Worksheet

The Origin worksheet looks superficially like a spreadsheet. However, it differs from a spreadsheet in some key respects:

- When working with spreadsheets, you typically refer to data points by intersecting row and column designations. In Origin, a worksheet column or ***dataset*** defines the primary data grouping while index (row) numbers serve to define a secondary grouping.
- While it is certainly possible to programmatically read and write to a single Origin worksheet "cell," a worksheet "cell" cannot be made to store mathematical formulae that refer to values contained in another cell as one frequently does when building a spreadsheet.

We will begin this tutorial by importing some sample data. Origin imports data stored in a wide variety of formats, and it will import very large files (>1 million lines of ASCII data) relatively quickly. These various file format import options are available from the worksheet's **File:Import** submenu.

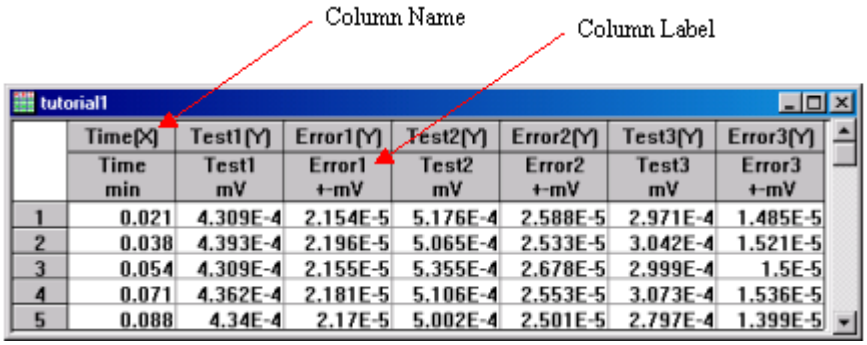


1. Click the **New Project** button  on the Standard toolbar. You do not need to save the results of the previous tutorial session. A new project file will open. The file should contain a single worksheet named **Data1**.
2. From the menu, choose **File: Import: Single ASCII…**. The Import ASCII dialog opens.
3. In the Origin \Tutorial folder, select Tutorial_1.DAT from the list of files. The data file is imported into the worksheet.

Note: You may need to resize the worksheet to see all columns.

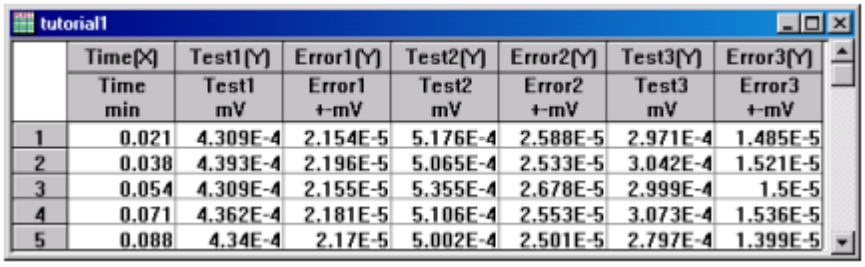
Notice that the worksheet is renamed using the name of the ASCII file that you imported. Recall that Origin does not allow special characters in window names so the underscore is dropped.

Also, notice that Origin has renamed the column headings. If you were to open Tutorial_1.dat with a text editor such as Notepad, you would see that the first two rows in the file contain text. In this case, Origin has used the first row of text to create the column name, while the first and second rows of text are used to create column labels. You can control the way that Origin treats the header information via the **ASCII Import Options** dialog box (**File:Import ASCII Options…**).



Designating Worksheet Columns as Error Bars

When you import data into a worksheet, Origin's default behavior is to designate the leftmost worksheet column as an X column and all columns to the right as Y columns. These X and Y designations are important because they determine the way that Origin will treat the data when they are plotted.



In this case, the Time column is designated as X and the Test*n* and Error*n* columns are designated as Y columns. In this worksheet the Error*n* columns are intended to be error measurements. Further, the Error1dataset is associated with the to Test1 dataset, the Error2 dataset is associated with the Test2 dataset, and so on. Column designations should be (from left to right) X, Y1, Y1 Error, Y2, Y2 Error, Y3, Y3 Error.

To specify the plot designation:

4. Position the cursor over the column heading labeled **Error1(Y)** and double-click with your mouse. The **Worksheet Column Format** dialog box for that column (dataset) opens.
5. In the **Options** group, select **Y Error** from the **Plot Designation** drop-down list and click **OK**.
6. Click the 

button at the top of the dialog.
7. The **Test2** column is properly designated as a Y column so we will skip to the next column.
8. Click the 

button.

9. To designate **Error2** as an error column, choose **Y Error** from the **Plot Designation** drop down list and click **OK**.
10. The **Test3** column is properly designated as a Y column, so we will skip to the next column.
11. Click the Next >> button and designate **Error3** as a **Y Error** dataset.

Our worksheet should now look like this:

tutorial1

	Time(X)	Test1(Y)	Error1(yEr±)	Test2(Y)	Error2(yEr±)	Test3(Y)	Error3(yEr±)
	Time	Test1	Error1	Test2	Error2	Test3	Error3
	min	mV	+-mV	mV	+-mV	mV	+-mV
1	0.021	4.309E-4	2.154E-5	5.176E-4	2.588E-5	2.971E-4	1.485E-5
2	0.038	4.393E-4	2.196E-5	5.065E-4	2.533E-5	3.042E-4	1.521E-5
3	0.054	4.309E-4	2.155E-5	5.355E-4	2.678E-5	2.999E-4	1.5E-5
4	0.071	4.362E-4	2.181E-5	5.106E-4	2.553E-5	3.073E-4	1.536E-5
5	0.088	4.34E-4	2.17E-5	5.002E-4	2.501E-5	2.797E-4	1.399E-5

We could also have set the plot designation for this particular worksheet in another, faster way – by selecting the entire worksheet, right-clicking and selecting **Set As: XYYErr** from the shortcut menu.

This tutorial is continued in the next section, **Plotting Your Data**.

Plotting Your Data

For most 2D and some 3D graph types, creating a graph is a simple, two-step process: (1) select your worksheet data and (2) click on a button on one of the plotting toolbars.

When you select and plot multiple datasets using this technique, Origin automatically groups the data plots and increments attributes such as the symbol type and color by dataset, so that you can easily distinguish individual plots.

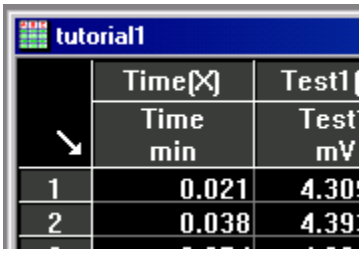
If you closed Origin after the last section, **Designating Worksheet Columns as Error Bars**, please refer back to it and complete steps 1 – 11 before continuing with this section.

Our tutorial 1 worksheet should look like this:

tutorial1

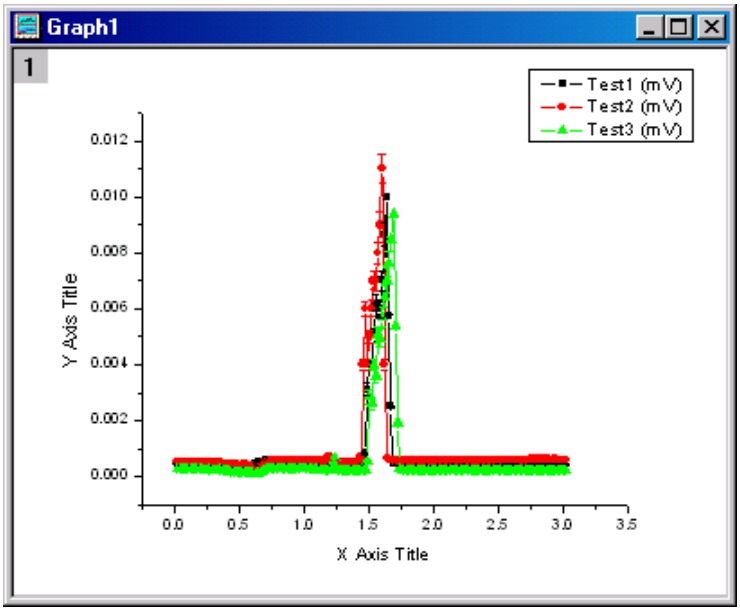
	Time(X)	Test1(Y)	Error1(yEr±)	Test2(Y)	Error2(yEr±)	Test3(Y)	Error3(yEr±)
	Time	Test1	Error1	Test2	Error2	Test3	Error3
	min	mV	+-mV	mV	+-mV	mV	+-mV
1	0.021	4.309E-4	2.154E-5	5.176E-4	2.588E-5	2.971E-4	1.485E-5
2	0.038	4.393E-4	2.196E-5	5.065E-4	2.533E-5	3.042E-4	1.521E-5
3	0.054	4.309E-4	2.155E-5	5.355E-4	2.678E-5	2.999E-4	1.5E-5
4	0.071	4.362E-4	2.181E-5	5.106E-4	2.553E-5	3.073E-4	1.536E-5
5	0.088	4.34E-4	2.17E-5	5.002E-4	2.501E-5	2.797E-4	1.399E-5

12. Position your cursor in the upper left corner of the worksheet. When the cursor changes to a downward pointing arrow, left-click once to select the entire worksheet.



13. Click the **Line+Symbol** graph button  on the 2D Graphs toolbar.

The data are plotted as a Line + Symbol plot, and the Error datasets are plotted as Y Error bars on the Test1, Test2, and Test3 data plots, respectively.

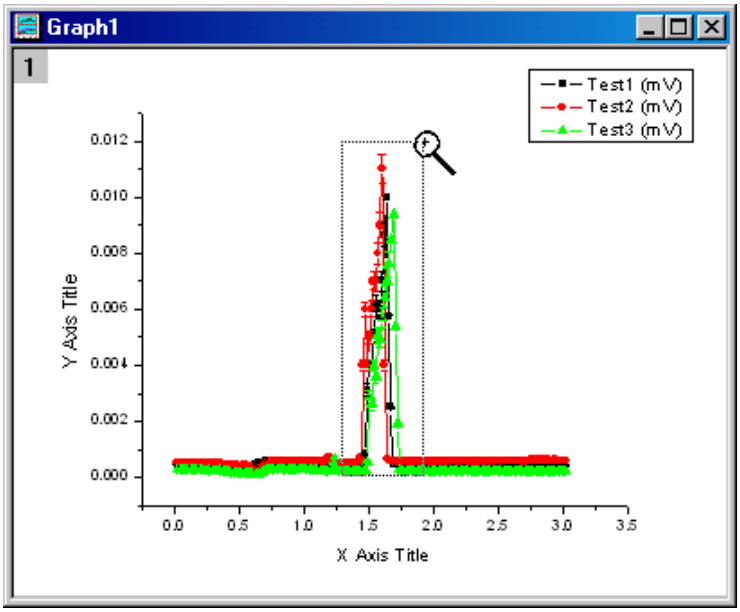


Focusing on a Region of Your Graph

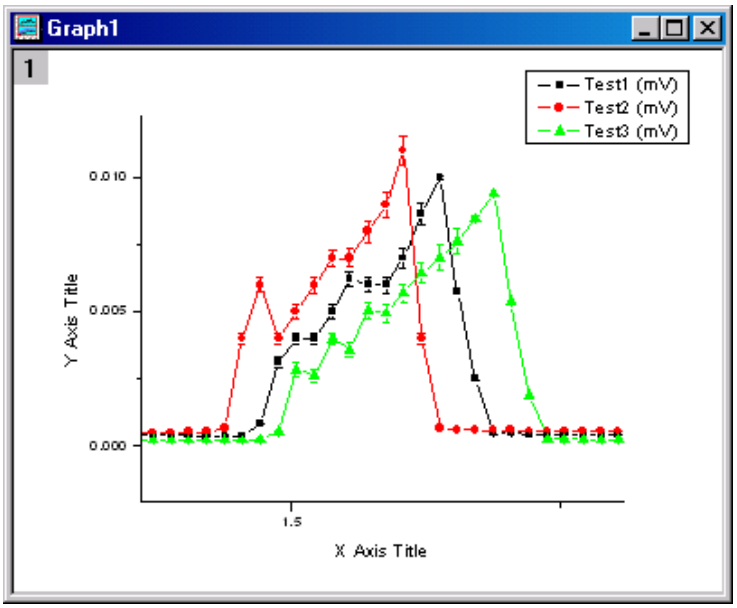
You may want to take a closer look at some portion of your plot(s), particularly if your plots are composed of large numbers of data points. Origin provides several tools to accomplish this, including the Enlarger tool. The Enlarger tool automatically rescales the axes of the graph to show only the region of the data plot(s) you select.

To enlarge a region of your data plots using the Enlarger tool:

14. Click on the **Enlarger** tool button  on the Tools toolbar. If your **Tools** toolbar is not visible, select **View:Toolbars** and enable the Tools toolbar.
15. Using the magnifying glass cursor, hold down the left mouse button and drag a box around the large peaks (near X = 1.5) in the graph window. Release the mouse button to complete the operation.



Your graph should now look something like this:



16. To replot the data over the entire range, click on the **Undo Enlarge** tool  on the Tools toolbar.

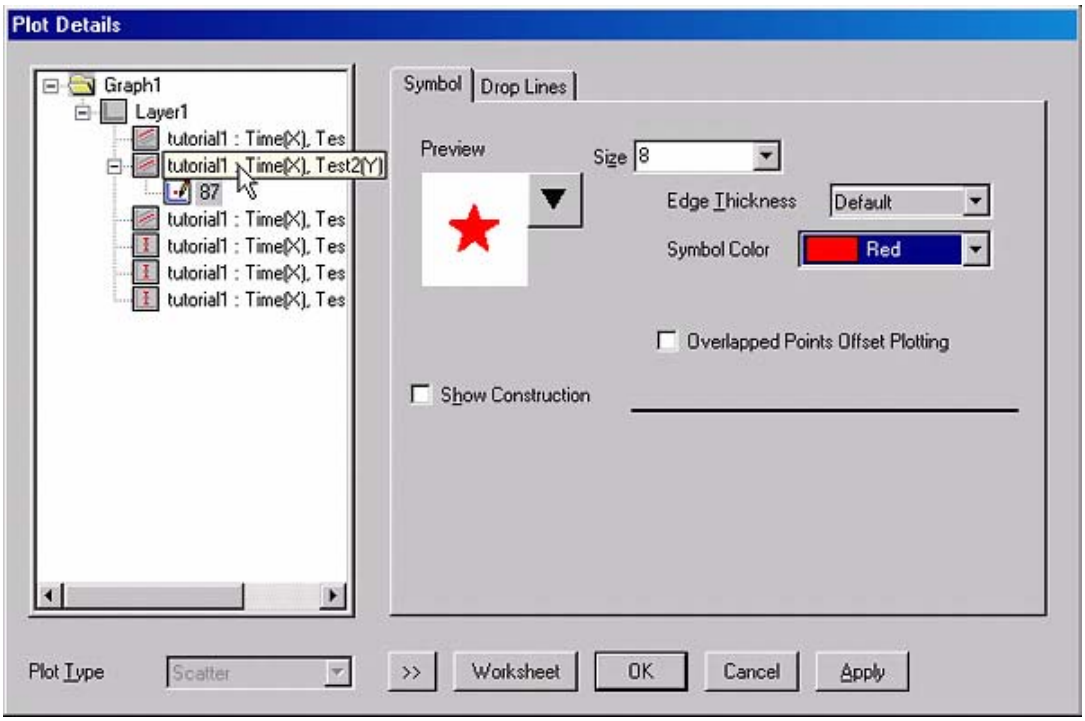
This tutorial is continued in the next section, **Customizing the Data Plot**.

Customizing the Data Plot

The flexibility to customize virtually every graphic element is one of the cornerstones of Origin's power. You can easily change the appearance of most graphic elements by pointing to that element and double-clicking with your mouse.

For most graphic elements, customizations are made from the Plot Details dialog box. This tabbed dialog allows you to customize every aspect of your graph with the exception of graph axes and text labels. Using

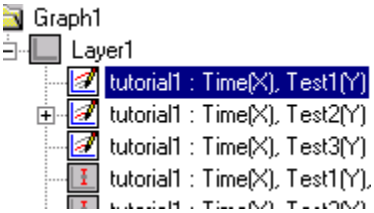
the Apply button, you can readily see how your changes will affect the appearance of your graph, without having to close and reopen the dialog when the result isn't to your liking.



As you look at the **Plot Details** dialog box, you can see that it is divided into two main parts: The left pane provides a hierarchical view of the graph's primary elements. The right side of the Plot Details dialog contains one or more tabs that provide access to the format options available for the page/layer/data set/data point selected in the left-hand pane.

To change the appearance of your graph using the Plot Details dialog:

17. Double-click the **Test1** data plot icon —■— in the graph legend. The **Plot Details** dialog box opens with the **tutorial1: Time(X), Test1(Y)** data plot icon selected in the left hand pane.



18. Select the **Symbol** tab on the right side of the Plot Details dialog box .
19. Click on the **Preview**  drop down list to open the symbol gallery and choose the **open circle** symbol type  from the drop-down list.
20. Click **OK**. The symbol type in the data plot will change from a filled square to an open circle.

You might have noticed that the other two data plots in the graph – Test2 and Test3 – *also had their symbol types changed*. Recall that when you select multiple datasets and plot them simultaneously as we did, Origin automatically groups the data sets (data plots) and increments attributes such as symbol type and

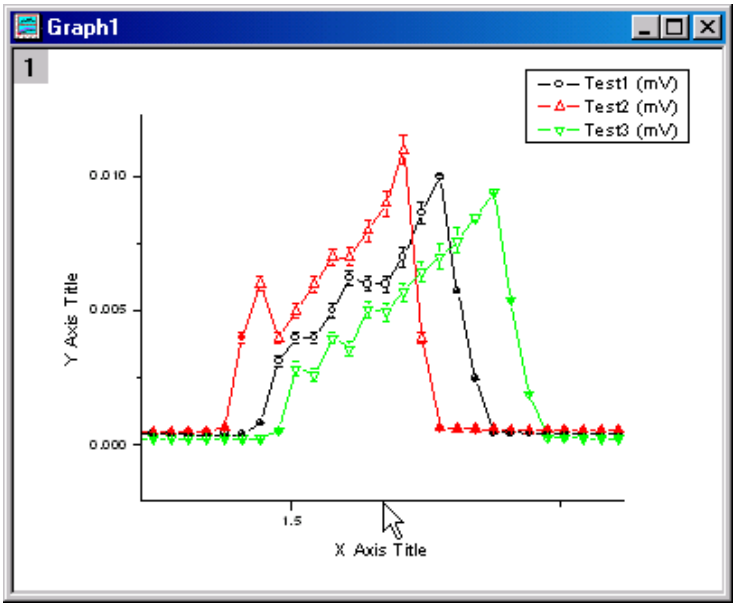
color by data set. While you can certainly specify your own individual plot attributes, this behavior allows you to quickly and easily plot multiple datasets while clearly distinguishing between them.

This tutorial is continued in the next section, **Customizing the Graph Axes**.

Customizing the Graph Axes

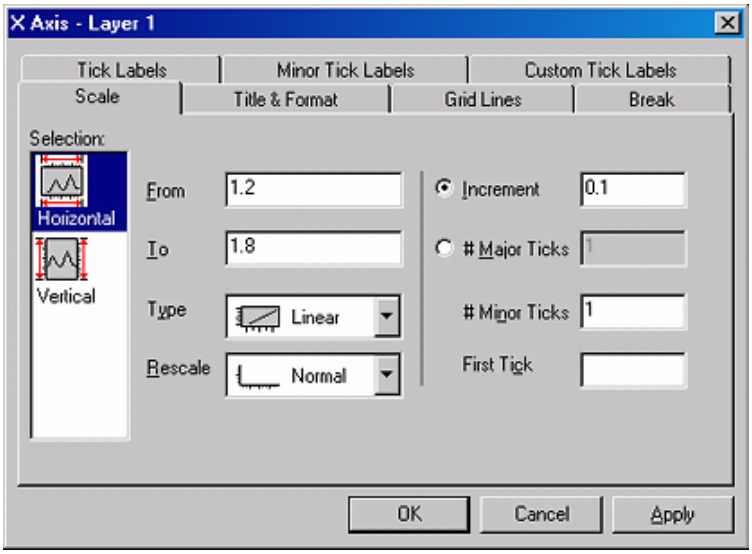
Double-clicking on a graph's axes opens the **Axis** dialog box. You can specify the axis you want to customize by selecting it from the **Selection** list box on the left side of the dialog box

To customize the graph axes using the Axis dialog box:



21. Double-click on the **X axis**. The **Axis** dialog opens.

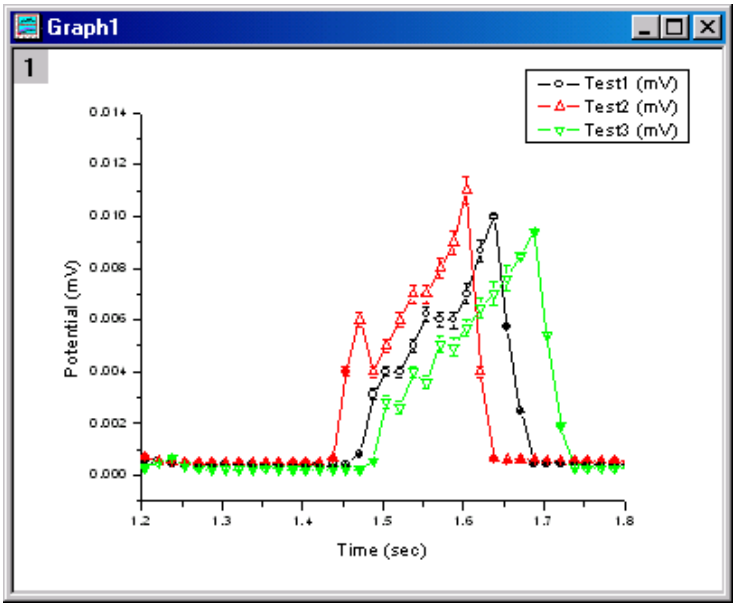
Note: If you double-click on the axis itself, the Axis dialog opens to the **Scale** tab. If you double-click on the axis *label*, the Axis dialog opens to the **Tick Labels** tab.



22. On the **Scale** tab, type **1.2** in the **From** text box, **1.8** in the **To** text box, and **0.1** in the **Increment** text box.
23. Select the **Title & Format** tab.
24. Type **Time (sec)** in the **Title** text box, overtyping the default text.
25. Select the **Left** icon 

from the **Selection** list box.
26. Type **Potential (mV)** in the **Title** text box, overtyping the default text.
27. Select the **Scale** tab.
28. Type **-.001** in the From text box, **.014** in the **To** text box, and **.002** in the **Increment** text box.
29. Click **OK**.

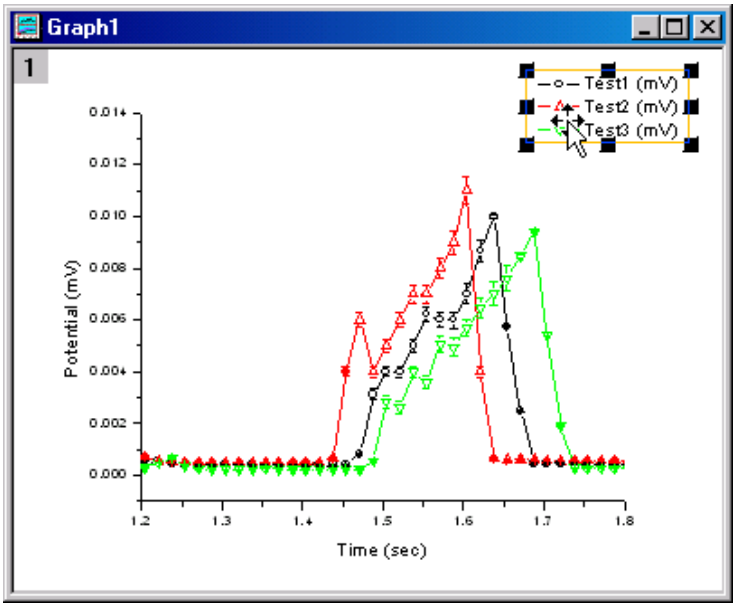
Your graph should now look something like this:



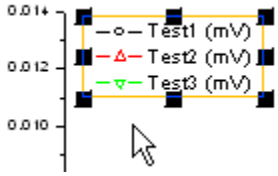
This tutorial is continued in the next section, **Adding Text to the Graph**.

Adding Text to the Graph

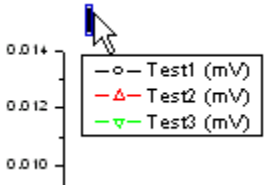
To further customize your graph, you can add annotations including text, arrows, lines, and shapes. The tools for adding these annotations are located on the Tools toolbar. Alternatively, you can right-click anywhere in the graph to Add Text using a shortcut menu.



30. First click and drag the legend so that it is located close to the Y axis. Click outside the legend to de-select it.

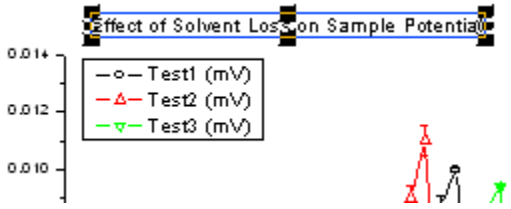


31. Click the **Text** tool button  on the Tools toolbar and then click in a blank area in the upper-left corner of the graph. A text label is created. You will see the cursor flashing at the text entry point If your **Tools** toolbar is not visible, click here.



32. Type **Effect of Solvent Loss on Sample Potential** in the text label. When you are done, click once outside the text label, or press **ESC**, to exit the text edit mode.

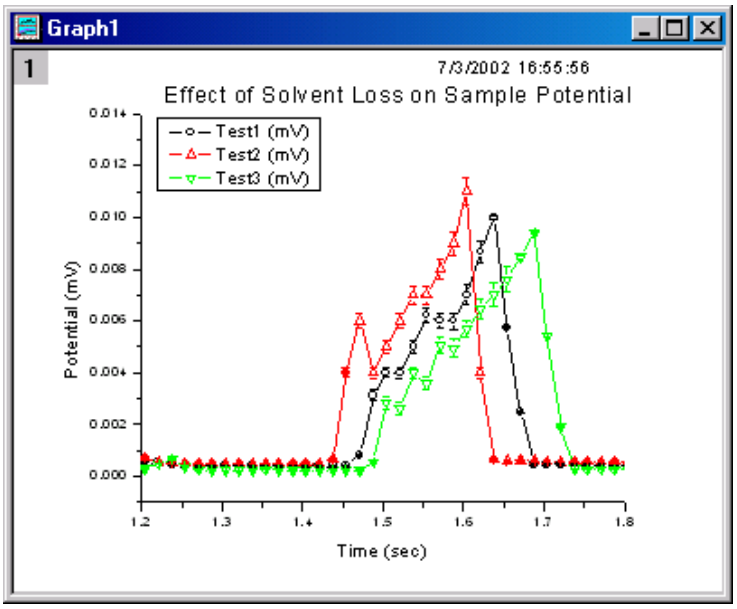
33. Click *once* on the new text label so that it becomes selected.



34. Drag the bottom right control handle to increase the size of the label.

35. When finished resizing, click outside the label to deselect it.

36. Click the **Date & Time** button  on the **Graph** toolbar to add a date and time stamp.

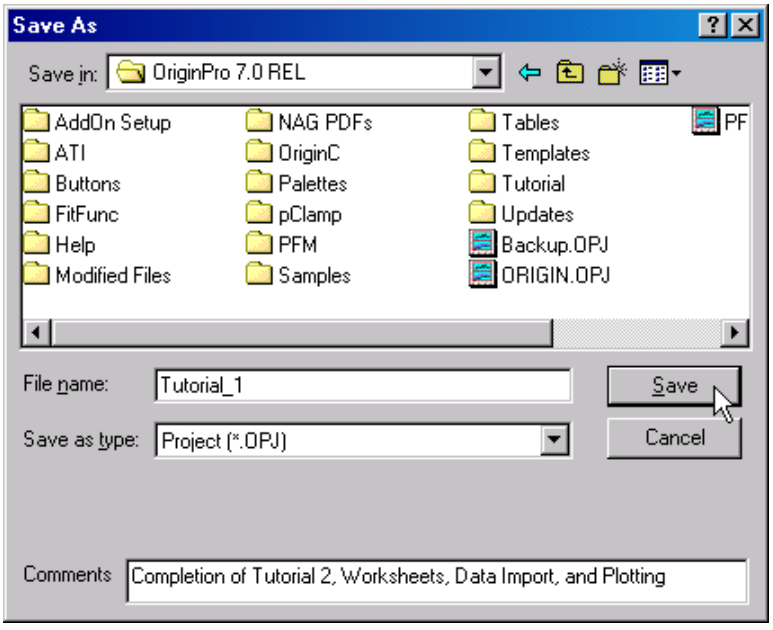


This tutorial is continued in the next section, **Saving Your Project**.

Saving Your Project

Your project currently consists of one worksheet and one graph window and the data displayed in both. Both of these windows and the data they contain are stored within the Origin project file when you save the project

37. Click the **Save** button  on the Standard toolbar.
38. Type **Tutorial_1** in the **File** name text box, then click **Save**. The project is saved as Tutorial_1.OPJ. Note the path to the saved file.



This concludes the tutorial on **Worksheets, Data Import, and Plotting**.

Tutorial 3, Working with Excel in Origin

You can open Excel workbooks directly in Origin, provided you have Excel 7 or later installed on your computer. This allows you to combine Excel's spreadsheet tools with Origin's powerful graphing and analysis tools.

This tutorial will show you how to open Excel workbooks in Origin, plot your Excel data using Origin graph templates, then save your Excel workbooks from Origin.

Opening an Excel Workbook in Origin

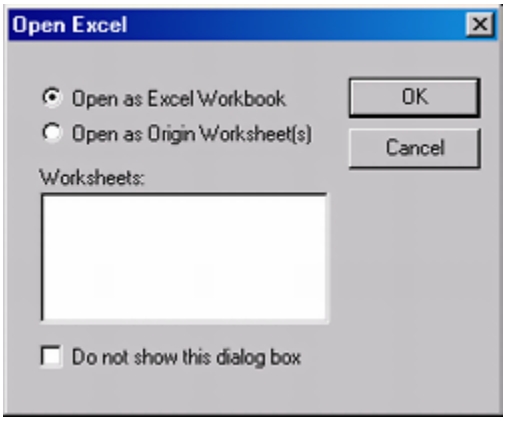
To follow this tutorial, you must have Excel version 7 or later installed on your computer, either as a local or network copy.

To open an existing Excel workbook:

1. On the Standard toolbar, click the **New Project** button  .
2. On the Standard toolbar, click the **Open Excel** button  .
3. In the Origin \Tutorial folder, select Tutorial_7.XLS from the file list.
4. Click **Open**.

The **Open Excel** dialog box opens. This dialog box gives you the choice of opening your Excel workbook (in Origin) as a workbook or opening it as an Origin worksheet. If you open a workbook as a workbook, you have access to Excel menus and toolbars, but somewhat restricted use of Origin's analysis and graphing features. If you open the workbook as an Origin worksheet, you have full access to Origin's graphing and analysis features, but you will not have access to Excel's spreadsheet features.

Note: The workbook/worksheet choice isn't entirely an "either/or" situation. You can open an Excel file twice – once as a workbook and a second time as a worksheet – and have access to everything that both programs have to offer. You can also copy and paste data from an Excel workbook into an Origin worksheet.



5. Select the **Open As Excel Workbook** radio button.
6. Click **OK**.

The workbook opens in the Origin workspace. Notice that a combination of Origin and Excel menus and toolbars are now displayed in the workspace.

Book1 - C:\Program Files\Origin Program Files\OriginPro7_Er347\Tutorial\Tutorial_7.xls

	A	B	C	D	E	F	G
1	Year	Domestic crude oil production (million barrels/day)	Crude oil imports (million barrels/day)	Petroleum products imports (million barrels/day)	Total imports (million barrels/day)	Crude oil exports (million barrels/day)	Petroleum products exports (million barrels/day)
2	1973	9.21	3.24	2.78	6.03	0.00	0.23
3	1974	8.77	3.47	2.42	5.89	0.00	0.22
4	1975	8.37	4.10	1.75	5.85	0.00	0.20
5	1976	8.13	5.28	1.81	7.09	0.00	0.22

Sheet1 Sheet2 Sheet3

Plotting an Excel Workbook in Origin

Plotting Excel workbook data in Origin is very similar to plotting Origin worksheet data. If a graph window is open, you can drag and drop data from a workbook into the graph window.

You can also create a new graph from your workbook data by (1) activating your workbook and (2) clicking a button on one of the plotting toolbars or by selecting a graph type from the Plot menu. An intermediary Select Data for Plotting dialog box opens to allow you to assign data sets to the graph.

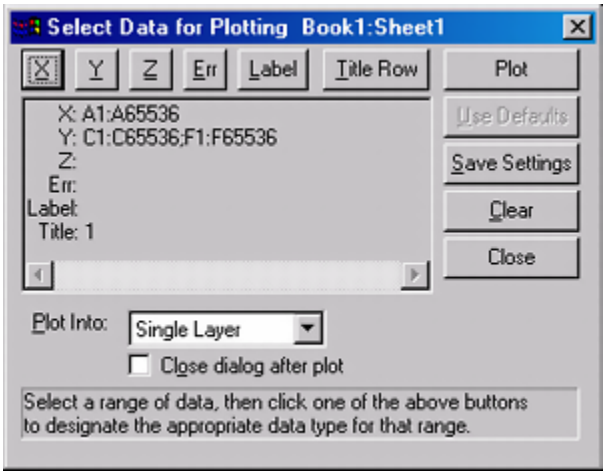
By default, the Select Data for Plotting dialog box opens when you select your Excel data and click a button or select a menu command to plot the data. After you become familiar with Origin's graph templates, you may be able to bypass this dialog box and plot your data directly from the workbook.

Creating a Graph Using the Select Data for Plotting Dialog Box

The **Select Data for Plotting** dialog box lets you select data and then assign plotting designations.

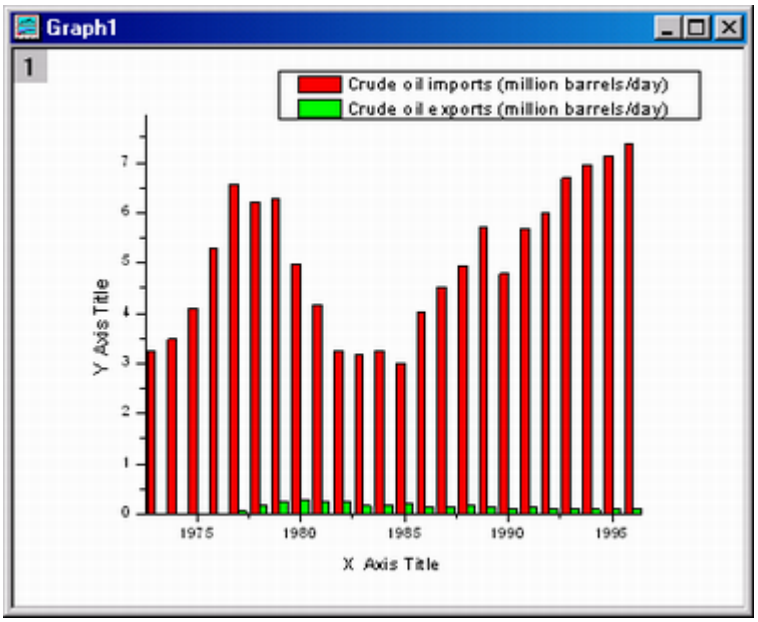
To create a graph using the Select Data for Plotting dialog box:

7. Click the **Column** button  on the 2D Graphs toolbar. This opens the **Select Data for Plotting** dialog box.
8. Reposition the dialog box so that you can see the column headings in the Excel workbook.
9. Click the column **A** heading in the workbook, then click  in the Select Data for Plotting dialog box.
10. Click the column **C** heading, then **CTRL+Click** on the column **F** heading and then click  in the Select Data for Plotting dialog box.

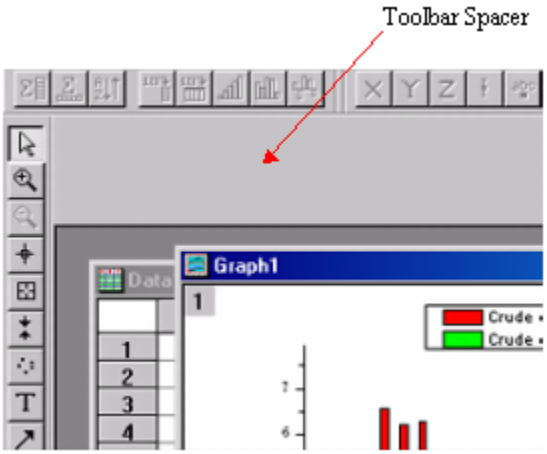


The box below the buttons indicates data ranges by plot designation. X:A1:A65536 means that the X data is being represented by column A, rows 1 to 65536. The last row number of the column is shown because you selected the whole column.

11. Click the **Plot** button in the **Select Data for Plotting** dialog box, then click **Close**.
12. Reposition the legend on the graph page.



When the graph window is active, you may notice a blank gray area in the toolbar region. This is called a toolbar spacer. The toolbar spacer is reserving space for the Excel toolbars, which will display when you re-activate the Excel workbook. If desired, you can hide the toolbar spacer by right-clicking in the spacer and selecting **Hide Toolbar Spacer** from the shortcut menu. When you re-activate the workbook, Origin will automatically display the spacer with the Excel toolbars.



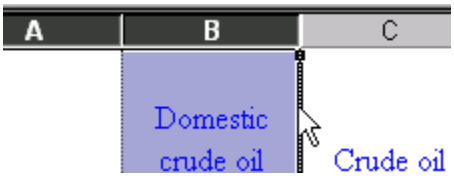
Creating a Data Plot by Dragging Data Into a Graph

When you drag Excel workbook data onto an Origin graph, Origin makes assumptions about the plotting designations of the selected data.

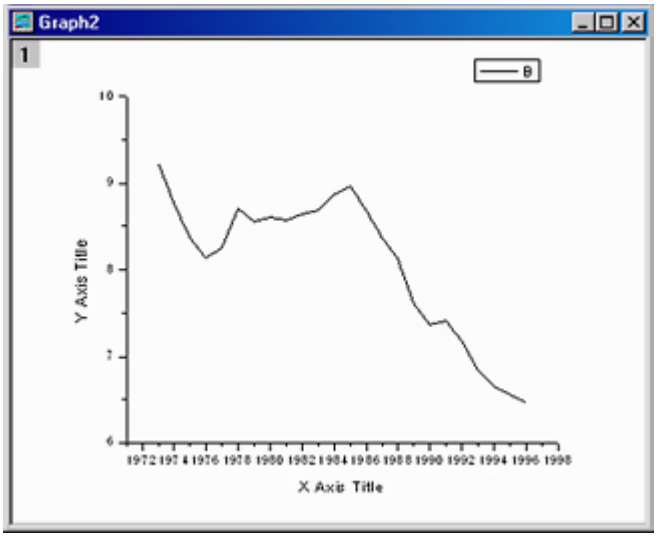
- If one column (or a range from one column) is highlighted, then this column supplies the Y values for the data plot. The data is plotted versus row number.
- If more than one column (or a range from more than one column) is highlighted, the leftmost column supplies the X values. All other columns supply the Y values. The data is plotted versus the X values.
- If more than one column (or range from more than one column) is highlighted and the CTRL key is depressed while dragging the data, then all the columns supply the Y values. The data is plotted versus row number.

To create a graph using the drag-and-drop method:

13. Click the **New Graph** button  on the Standard toolbar. Reposition/resize the Graph2 window so that when the workbook is active, it doesn't completely hide the new graph window.
14. Select **Window:Book1** to activate the Excel workbook window.
15. Highlight the first two columns in the workbook. Note that as you move your mouse over the right edge of your two columns of selected data, the cursor changes.



16. Point to the right edge of the B column and, when the cursor becomes a pointer, hold down the left mouse button and drag the data to the Graph2 window. When you release the mouse button, column B is plotted against column A as a line plot in the graph window.

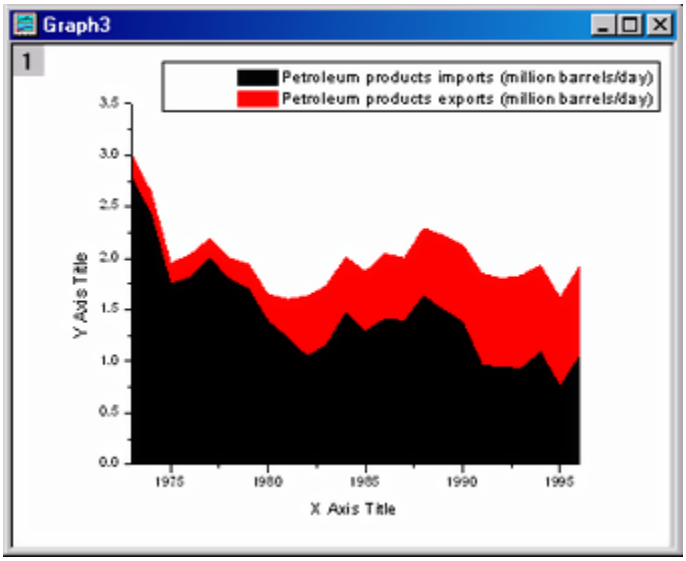


Creating a Graph Using Origin's Default Plot Assignments

This plotting method allows you to select your workbook data and graph type, and then Origin creates the data plots by making assumptions about the plotting assignments of the selected data. This plotting method is not available by default. It must be activated from the Options dialog box (**Window:Origin Options** when a workbook is active; **Tools:Options** when an Origin window is active).

To create a graph using Default Plot Assignments:

17. Select **Tools:Options** to open the **Options** dialog box. (If the workbook is active, then select **Window:Origin Options**).
18. On the **Excel** tab, select the **Default Plot Assignments** check box.
19. Click **OK**, then click **No** (no need to save as a start up option) to the dialog box that opens.
20. Make the Excel workbook active.
21. Click on the column A heading, then **CTRL+click** on the column D and G headings.
22. Click the **Area** graph button  on the 2D Graphs toolbar.
23. Reposition the graph legend.



Saving an Excel Workbook in Origin

There are two ways to save Excel workbooks with your Origin projects. You can save the workbook **internal** to the project, which means that it is saved as part of the Origin project and can only be opened by opening the Origin project file.

The other choice is to save the workbook as **external** to the Origin project file. When you save a workbook externally, a link to the workbook is maintained in the project file.

The chief advantage to storing your workbook internally is that all of your graphs, analyses, support data, etc., can be bundled together in a single file. This makes it easy to share data with other Origin users.

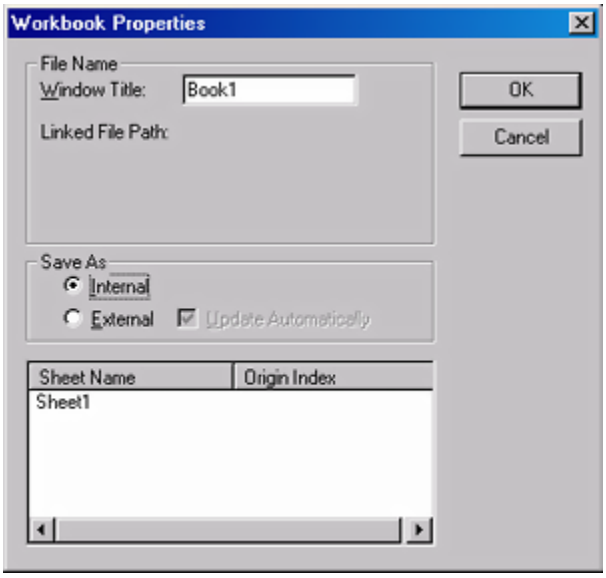
Of course, sometimes sharing data with colleagues may mean that you are working on a common file that is saved to a shared folder on the your network. In such cases, saving your data to an external Excel workbook file may be the way to go.

Note: Internally saved workbooks do not maintain links with external Excel workbook files. Thus, if you save a workbook internally while colleagues continue to work on the external Excel file, your project workbook will not be updated.

By default, pre-existing workbooks that are opened in an Origin project are saved external to the project, whereas workbooks that are created from within Origin are saved internal to the project. However, the save option can be changed in the **Workbook Properties** dialog box.

To save the workbook internal to the Origin project:

24. Make the Excel workbook active.
25. Right-click on the title bar of the workbook and select **Properties** from the shortcut menu. This opens the **Workbook Properties** dialog box.
26. Select the **Internal** radio button in the **Save As** group.



27. Click **OK**.
28. Select File:Save Project As.
29. Type **Excel_Tutorial** in the File Name text box.
30. Click **Save**.

The project is saved and the workbook is saved internal to the project. Any changes that you now make to the Excel workbook located in the \TUTORIAL folder will not be reflected in the project file the next time it is opened.

This concludes the **Excel** tutorial.

Tutorial 4, Exploring your Data

Origin offers a number of data selection tools to aid in exploring and analyzing your data; these include the Data Reader, Screen Reader, Data Selector, Enlarger, Zoom, Region of Interest and the various data masking tools available from the Mask toolbar. To learn more about these features, consult the Origin help file at **Help:Origin…**.

In this tutorial, you will use the masking tools to selectively exclude unwanted data points from your analyses. You will also mathematically transform some worksheet data, do a primary and secondary sort of some worksheet data, plot a range of data and perform a quick linear fit on a plotted data set.

Importing Data

To begin this tutorial, you will open a new Origin project and import ASCII data. To import an ASCII data file:

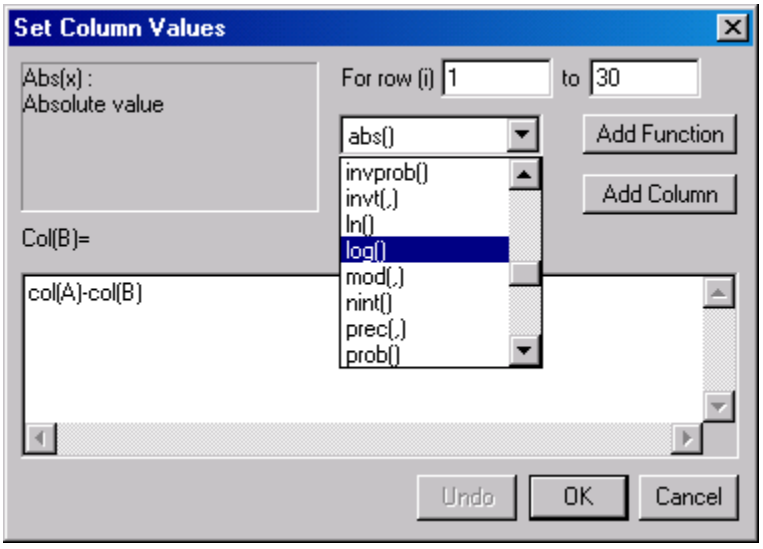
To import an ASCII data file:

1. Click the **New Project** button on the Standard toolbar.
2. Click the **Import ASCII** button, then select Tutorial_2.DAT from the Origin \Tutorial folder.
3. Click **Open**.

Note: A portion of the worksheet (and the imported data) may not be visible. You may need to use the scroll bars at the bottom and right edges of the worksheet to locate worksheet columns of interest.

Transforming Column Values

You can create or transform data sets with a mathematical expression via the **Set Column Values** dialog box. This dialog box provides a text box for you to type a value or mathematical expression to apply to a entire column, or portion of a column, of worksheet data .

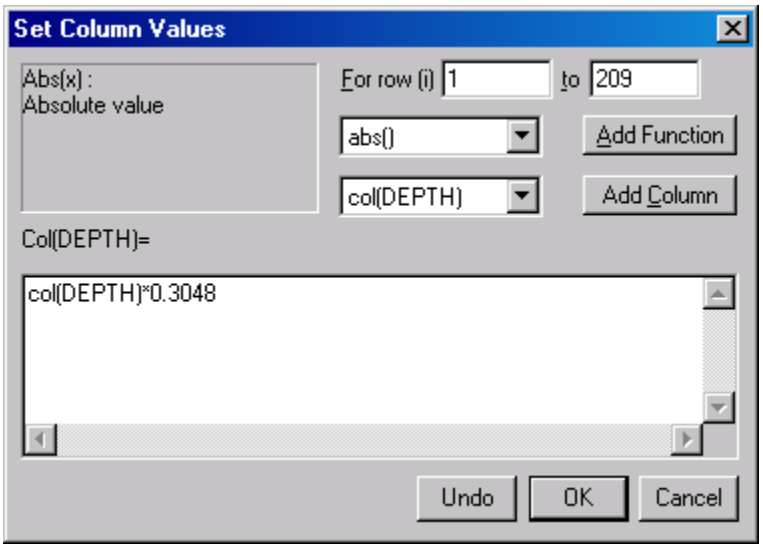


The **Set Column Values** dialog also includes a function drop-down list from which you can select a function to add to the text box. A column drop-down list contains a list of all columns in the active worksheet. Select the column you want to add to the text box, then click **Add Column** to add the selected column to the text box.

Note: It is perfectly acceptable to type column names into the text box. Just use the col(*colname*) syntax to refer to a worksheet column.

To transform a column of values via the Set Column Values dialog:

4. Right-click on the **Depth(Y)** column heading.
5. Select **Set Column Values…** from the shortcut menu.
6. Leave col(A)-col(B) highlighted in the text box and select **col(DEPTH)** from the **Add Column** drop-down list.



7. Click **Add Column**. **Col(DEPTH)** overwrites the highlighted text.
8. Leave the cursor at the current location in the text box and type ***0.3048** in the text box. This will have the effect of multiplying the current set of values in col(DEPTH) by 0.3048.
9. Click **OK**. Note that the column of values has been transformed.

Sorting Worksheet Data

Origin can sort data in individual columns, multiple columns, or entire worksheets. Origin offers simple sorting in which selected data are sorted using a "sort by" column, in ascending or descending order.

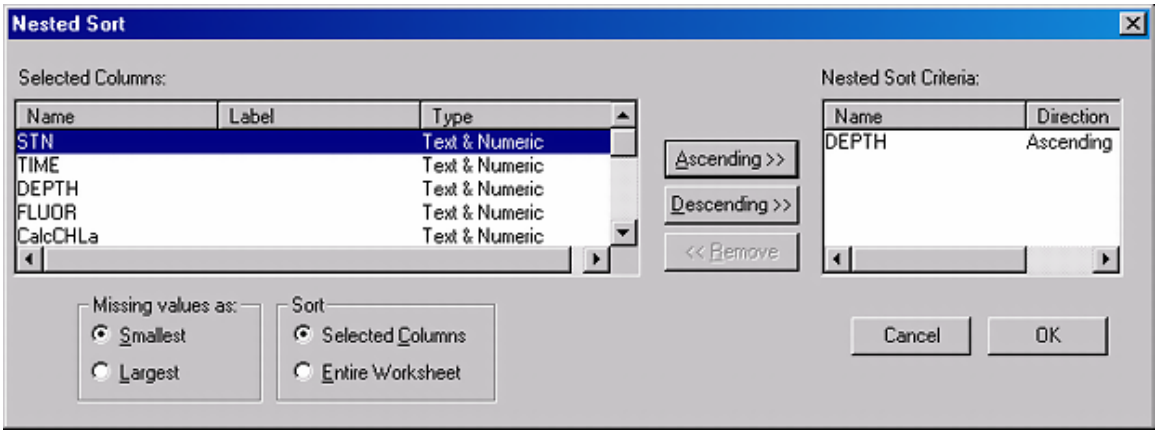
Origin also offers custom sorting of multiple columns, in which sorting is done first on a primary column, while subordinate columns are sorted in nested fashion in either ascending or descending order.

To sort worksheet data:

10. Position your cursor in the upper left corner of the worksheet. When the cursor changes to a downward pointing arrow, left-click once to select the entire worksheet.
11. Click the **Sort** button  on the **Worksheet Data** toolbar to open the **Nested Sort** dialog box.

12. Select **DEPTH** from the **Selected Columns** list box, then click the Ascending >> button. The column is added to the **Nested Sort Criteria** list box. This selection makes DEPTH the primary column, to be sorted in ascending order.

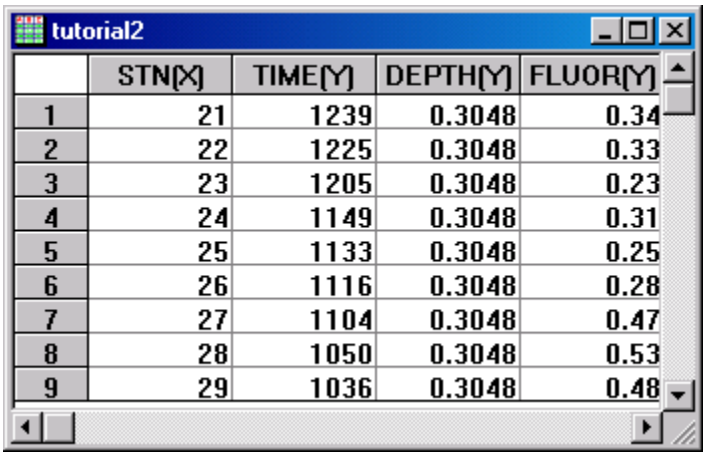
13. Select **STN** from the **Selected Columns** list box, then click Ascending >>. This makes STN the secondary sort column in ascending order.



14. Click **OK**.

15. De-select the worksheet by clicking once in the upper–left hand corner of the worksheet (without the cursor changing to the downward pointing arrow).

The entire worksheet is sorted so that the values in the DEPTH (primary) column are ascending. For duplicate values in the DEPTH column, the values in the corresponding rows of the STN (secondary) column determine how the worksheet data are ordered.



	STN(X)	TIME(Y)	DEPTH(Y)	FLUOR(Y)
1	21	1239	0.3048	0.34
2	22	1225	0.3048	0.33
3	23	1205	0.3048	0.23
4	24	1149	0.3048	0.31
5	25	1133	0.3048	0.25
6	26	1116	0.3048	0.28
7	27	1104	0.3048	0.47
8	28	1050	0.3048	0.53
9	29	1036	0.3048	0.48

Note: To sort the values in *every* column in the worksheet in ascending order, you must sort columns individually. Otherwise, the worksheet is sorted on the primary column and all values in row *ii* will still be in a common row after data are sorted.

Plotting a Range of the Worksheet Data

You can delimit your worksheet data so that subsequent plotting and analysis are performed only on a subset of your data.

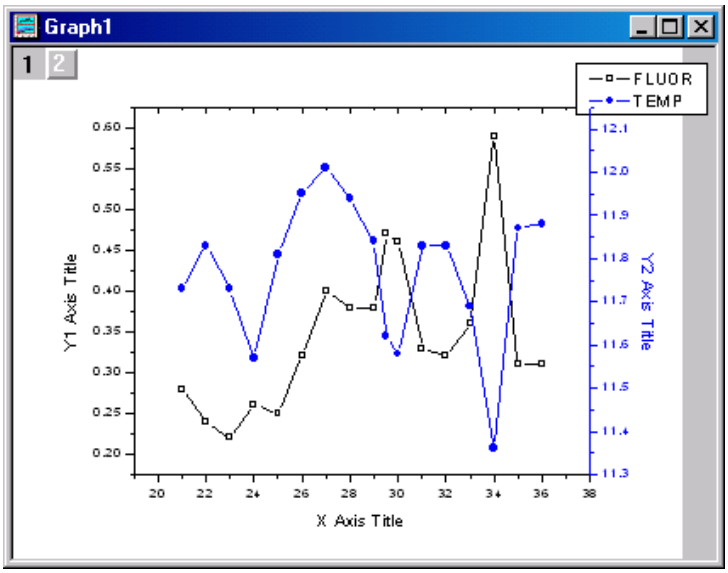
To select a range of worksheet data:

16. From the menu, select **View:Go To Row**.
17. Type **52** in the **GoTo Row** dialog box, and click **OK**.
18. Right-click on the row heading of row number 52.
19. Select **Set As Begin** from the shortcut menu.
20. Use the vertical scroll bar to move down in the worksheet so that row number 68 is visible.
21. Right-click on the row heading of row number 68.
22. Select **Set As End** from the shortcut menu.

Notice that the data outside the selected range is no longer displayed in the worksheet. The data *has not been deleted* from the worksheet, *only hidden* to provide for easier viewing of the selected range.

To plot this subset of data:

23. Click on the **FLUOR(Y)** column heading to select the column.
24. Scroll to the right in the worksheet, hold down the **CTRL** key and click on the **TEMP(Y)** column heading. This selects the both the TEMP and the FLUOR columns.
25. Click the **Double Y Axis** button  on the 2D Graphs Extended toolbar (or select **View:Toolbars** if the 2D Graphs Extended toolbar isn't open.).



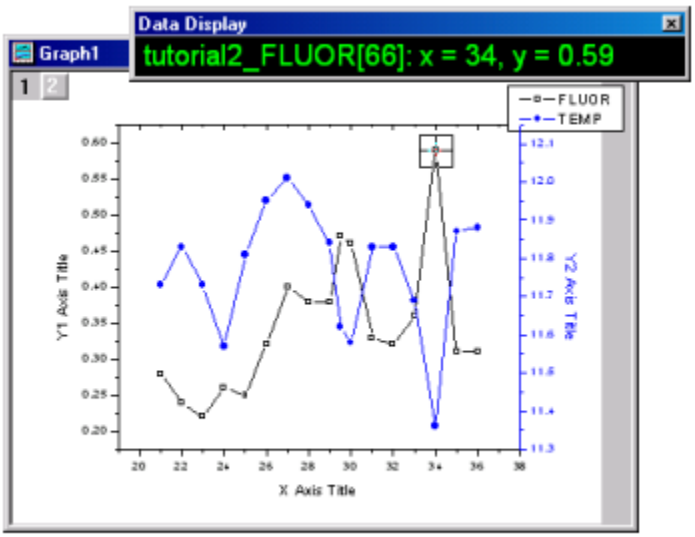
Masking Data in the Graph

The **Mask** toolbar provides other means for excluding data from Origin's analysis and fitting routines. You can mask individual data points or a range of data. Once data are masked, you can change the masked data color, hide or show the masked data, swap the masked and unmasked data, and enable or disable masking.

To mask a data point in the graph:

26. Click the **Mask Point Toggle** button  on the Mask toolbar. This activates the **Data Reader** tool.

27. Move the cursor (it is now a cross-hair) to the Double Y-axis plot that we just created and left-click anywhere on the FLUOR data plot (the black, open square symbols) in your graph window.
28. Using the right or left arrows on the keyboard, move the cursor to the data point at **X = 34** and **Y = .59**. The **Data Display** tool displays the point coordinate values.



29. Press **ENTER** or double-click to mask this point.
30. Click the **Change Mask Color** button on the Mask toolbar. The color of the masked data point changes to green. If you scroll the worksheet to find the data point that you just masked in the graph, you will see that it now displays with a green background to indicate that this particular data point is masked.

	STN(X)	TIME(Y)	DEPTH(Y)	FLUOR(Y)
59	28	1050	1.2192	0.38
60	29	1036	1.2192	0.38
61	29.5	1024	1.2192	0.47
62	30	1010	1.2192	0.46
63	31	953	1.2192	0.33
64	32	944	1.2192	0.32
65	33	923	1.2192	0.36
66	34	908	1.2192	0.59

31. With the graph window active, click the **Hide/Show Masked Points** button on the Mask toolbar to hide the masked data point. Note that the point is removed from the graph.
32. Once again, click the **Hide/Show Masked Points** button on the Mask toolbar . This restores the masked data point to the graph.

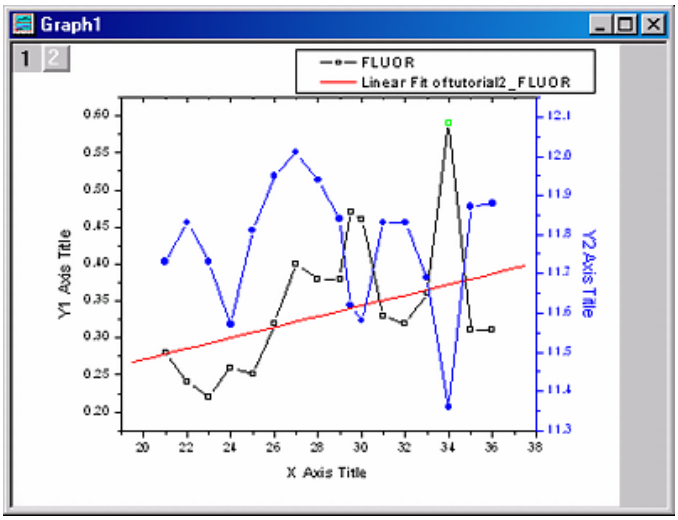
Performing a Linear Fit on the FLUOR Data Plot

Masking the data point in the FLUOR data plot excludes that point from analysis and fitting operations. You can, however, disable the mask on the data point, and analyze or fit all the data points in the current selection range.

To perform a linear regression with the data point masked:

33. Select **Analysis:Fit Linear**. A "best-fit" line is added to the graph and the **Results Log** opens to display the fitting results (parameter and error value estimates, R-value, SD, etc.). You can scroll the Results Log to view the results.

34. Click the **Refresh** button on the Standard toolbar and reposition the legend so it fits on the page (remember that the page is the white area within the graph window; anything that laps onto the gray area will not be visible when the graph is printed or exported).




```
[7/9/2002 15:35 "/Graph1" (2452464)]
Linear Regression for tutorial2_FLUOR:
Y = A + B * X

Parameter      Value          Error
------------------------------------------------------------
A              0.12658        0.11156
B              0.00723        0.00391
------------------------------------------------------------

R              SD             N              P
------------------------------------------------------------
0.44343        0.06872        16             0.08537
```

By default, the **Results Log** shows the results for all curve fitting done in the *active* Project Explorer folder. To change this default behavior, right-click in the Results Log and select a different reporting option.

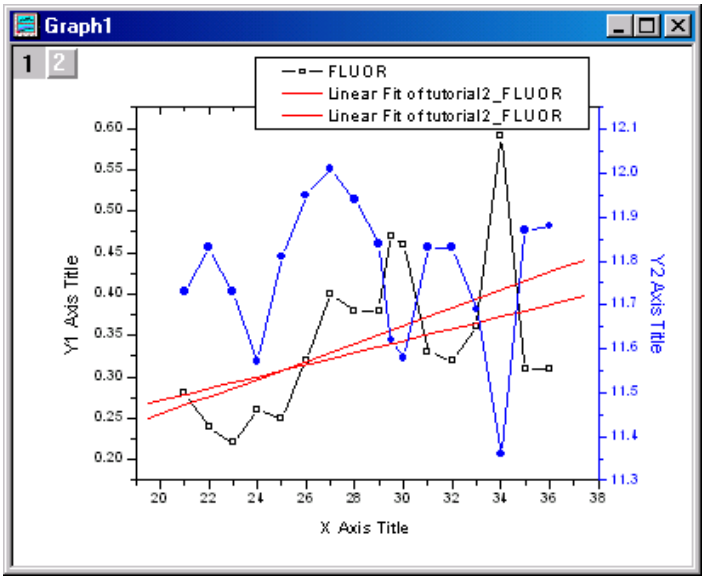
Each time a fit is performed, the results are appended to the Results Log. Each entry in the Results Log includes a date/time stamp, the window name, a numeric stamp which is the Julian day, the type of analysis, and the results.

To perform a linear fit with the Mask disabled:

35. Click the **Disable/Enable Masking** button on the Mask toolbar. The masked data point changes from green to black.

36. From the menu, select **Analysis:Fit Linear**. A second regression line is drawn on the graph.

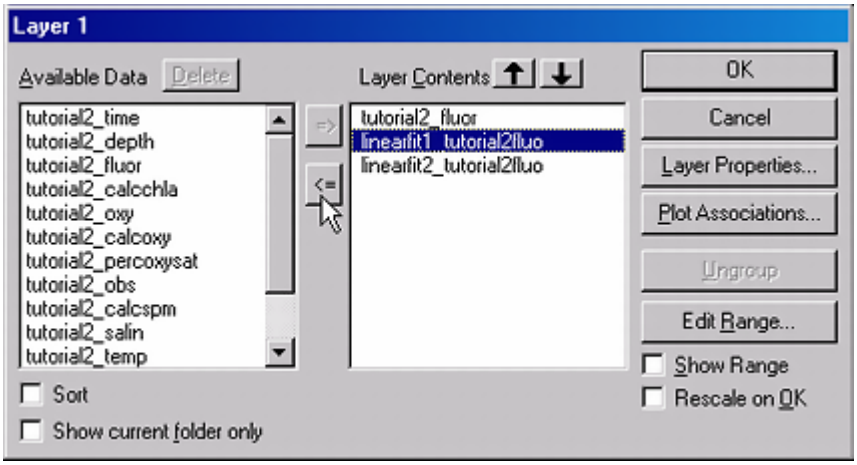
37. Verify that the mask is disabled by comparing the two fit results in the Results Log.



To remove the initial regression line from the graph:

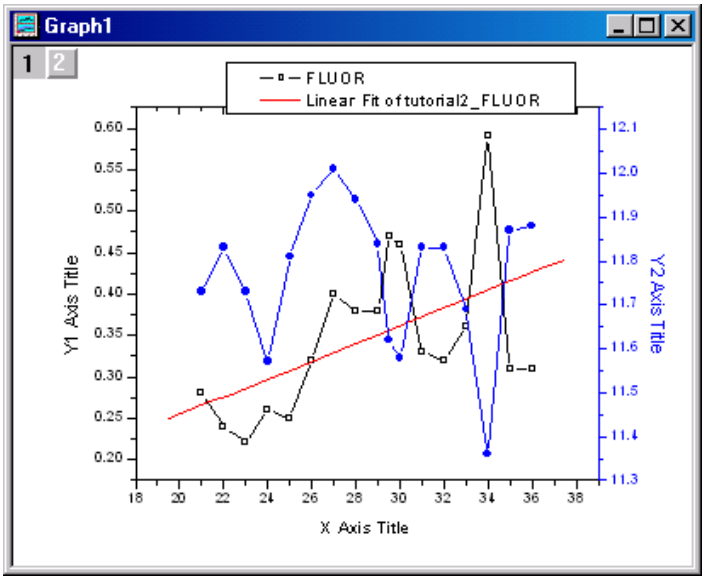
38. Double-click on the Layer 1 icon 

in the upper-left corner of the graph. This opens the Layer 1 dialog box.
39. Select linearfit1_tutorial2fluo from the Layer Contents list box.



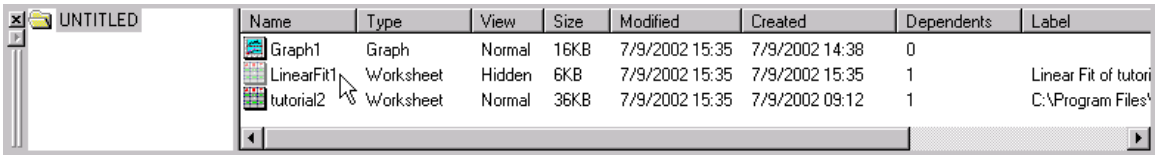
40. Click the 

button to remove linearfit1_tutorial2fluo from the **Layer Contents** list box.
41. Click **OK** to close the dialog box.
42. To update the legend, click the **New Legend** button  on the Graph toolbar. You may also need to click the **Refresh** button  on the Standard toolbar.



Saving the Project

Your Origin project currently contains your data, worksheets, graph, analysis results and the current folder organization as it shows in Project Explorer.



To save the current Origin project:

43. From the menu, select **File:Save Project**.
44. Type a name in the **File Name** text box.
45. Click on the **Save** button.

This concludes the **Exploring Your Data** tutorial.

Tutorial 5, Creating Multiple Layer Graphs

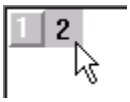
The layer is the fundamental building block of Origin graphs. A layer is an Origin object consisting of one set of controlling axes. You may opt to hide or show any or all of the four axes (top X, bottom X, right Y, and left Y) that make up the layer. A layer contains text labels, drawing objects, or data plots, that may be shown or hidden. A graph window must contain at least one layer and may include as many as 80 layers.

This tutorial will introduce you to the graph layer concept. In addition, you will learn about Origin's built-in multiple layer graph templates. You will also learn how to create your own multiple layer graph and save it as a template for later use.

The Graph Layer Concept

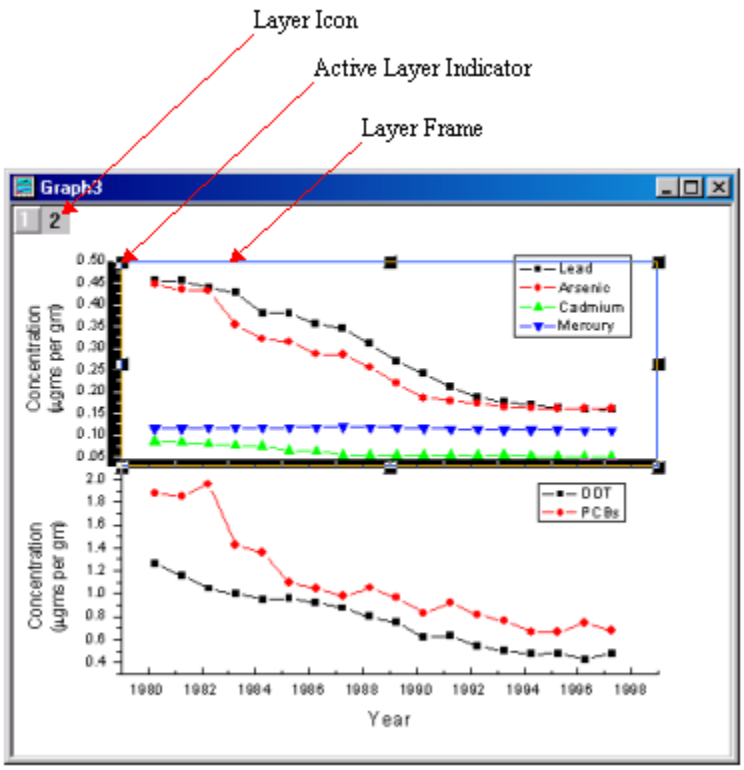
There is always one, and only one, active layer in a graph window. This active layer is the only layer that is subject to program commands. Data or objects added to a graph can only be added to the active layer.

Unless intentionally hidden, a graph window will have one or more layer icons – one for each layer in the graph window – in the upper left corner of the graph window. The active layer is denoted by a depressed layer icon.



- To hide the layer icons, select **View:Show:Layer Icons**.
- To highlight the axes in the active layer, select **View:Show:Active Layer Indicator**.

Reminder: Just as there can be only one active layer within a graph window, recall that there can be only one active window within the Origin workspace. Thus, the above actions only affect the active graph window.



You can activate a layer in several ways:

- Click once on the corresponding layer icon in the upper left corner of the graph window.
- Click once anywhere within the graph's layer frame
- Click once on any object (text, axis titles, etc.) that is part of or is attached to the layer.

Opening the Project File

The data for this tutorial is provided in an Origin project file.

To open the Origin project file:

1. Click the **Open** button  on the Standard toolbar.
2. In the Origin\Tutorial folder, select Tutorial_3.OPJ from the list of files.
3. Click **Open**. A project containing four graph windows and a worksheet opens.

Note: Three graph windows and the worksheet are in a minimized state at the bottom of your Origin workspace.

Origin's Multiple Layer Graph Templates

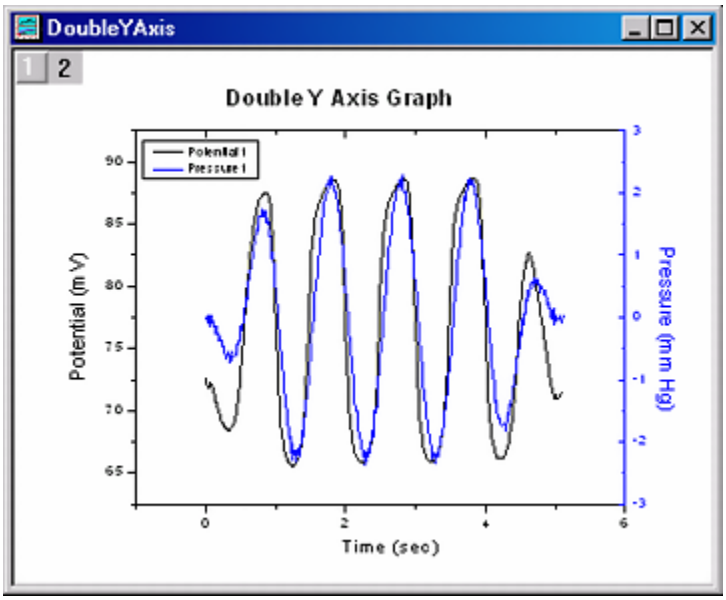
Origin ships with a number of highly useable, built-in, multiple layer graph templates. Putting one of these templates to use is a two step process: (1) first, you to select a range of data, then (2) you click a button to plot the selected data into multiple layers in a graph window.

The trick is to become comfortable with the behavior of grouped datasets, the graph layer concept, and the workings of the various multiple layer templates. Once you accomplish this, you will be well on your way

to churning out dozens of visually appealing and highly illustrative multi-layer graphs with just a few clicks of the mouse.

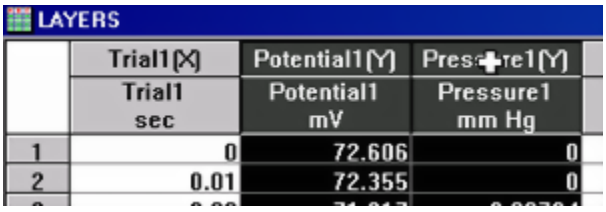
The double Y axis graph template is ideal for plotting data that consists of two or more dependent data sets and a common independent data set. A sample double Y axis graph is currently active in your project.

Note that, unlike the example in our lead-in discussion of the graph layer concept, the layers in this graph are superimposed. This is an important point and one that helps to illustrate the power of Origin's layer concept. Because layers are constructed independently, you can combine and manipulate them in innumerable ways. This allows a great deal of flexibility in the graph creation process.



Origin's built-in horizontal 2 panel graph template is ideal for plotting related data that do not share an independent data set. Tutorial_3.OPJ contains an example of a graph made using this template.

4. Click on the **Restore Up** icon on the minimized **Horizontal2Panel** window icon at the bottom of your workspace. The Horizontal2Panel graph should now be visible.
5. Click on the **Restore Up** icon on the minimized **Layers** worksheet at the bottom of your workspace. The Layers worksheet should now be visible.
6. Select the **Pontential1(Y)** and **Pressure1(Y)** columns by holding down your left mouse button and dragging the "+" cursor across both column headings. When both are selected, release the mouse button.



To plot the selected data using Origin's Horizontal 2 Panel template:

7. Click the **Horizontal 2 Panel** button on the 2 D Graphs Extended toolbar.

The resulting graph should look very much like the Horizontal2Panel graph window, minus some text annotations.

8. Click on the **Close** icon  of the graph window that you just created. You may delete the window.

Origin's vertical 2 panel graph template provides the same data presentation as the horizontal 2 panel graph template, but in a one column/two row configuration (i.e. the graph panels are stacked one on top of the other).

9. Click on the **Restore Up** icon  on the minimized **Vertical2Panel** window icon at the bottom of your workspace. The Vertical2Panel graph should now be visible. Resize/move this window so that you can clearly see the Layers worksheet.

To plot the selected data using Origin's Vertical 2 Panel template:

10. Select the **Potential1(Y)** and **Pressure1(Y)** columns of the Layers worksheet if they are not selected and click the **Vertical 2 Panel** button  on the 2D Graphs Extended toolbar.

The resulting graph should look very much like the Vertical2Panel graph window, minus some annotations.

11. Click on the **Close** icon  of the window that you just created. You may delete the window.

When your worksheet includes multiple X columns, Y columns in the worksheet plot against the nearest X column to the left. Though this default behavior can be disregarded by selecting non-contiguous columns with **CTRL** selection, the default plotting behavior allows you to quickly create graphs from associated XY data sets.

The Layer worksheet contains a second column that should be designated as an X column – this is the Trial2(Y) column – so we will change the plot designation to **X** so that we can proceed with our last multiple layer template plotting exercise.

To designate a second X column:

12. Right-click on the **Trial2(Y)** column heading and select **Set As:X** from the shortcut menu, or double-click on the column heading and choose **X** from the **Plot Designation** drop-down list in the **Worksheet Column Format** dialog box.

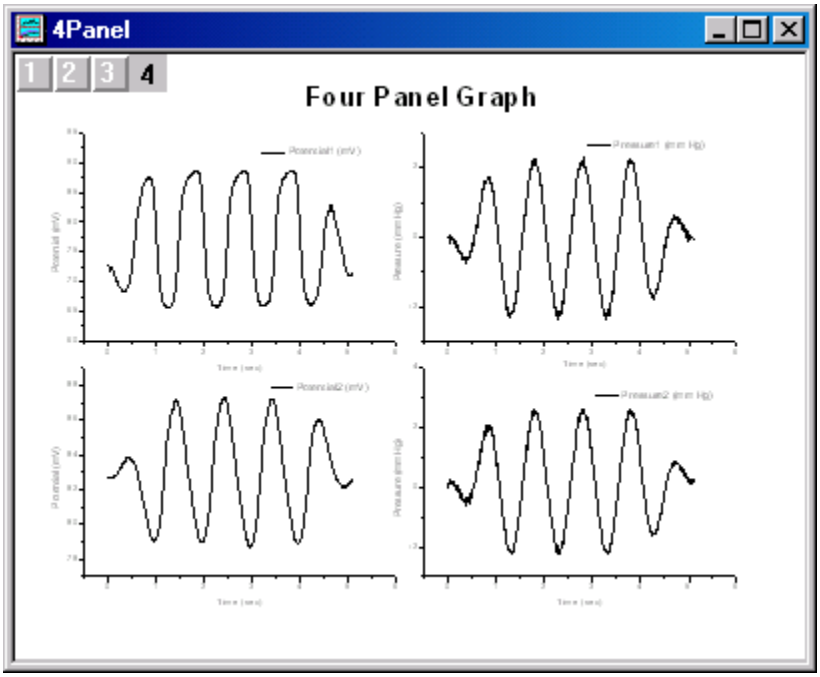
The Trial2 column designation changes to **X2** and the columns to the right of it are designated as **Y2**. In addition, the Trial1 column designation changes to **X1** and the columns between Trial1 and Trial2 are designated as **Y1**. Thus configured, the default behavior is for **Y1** columns to plot against the **X1** column, **Y2** columns to plot against the **X2** column, etc.

To create a multiple layer graph using the 4 Panel template:

13. Select the entire worksheet by placing the cursor in the upper left corner of the worksheet. When the cursor becomes a downward pointing arrow, left-click once.

14. Click on the **4 Panel** button  on the 2D Graphs Extended toolbar.

Your 4 layer, 4 panel graph should look something like this:



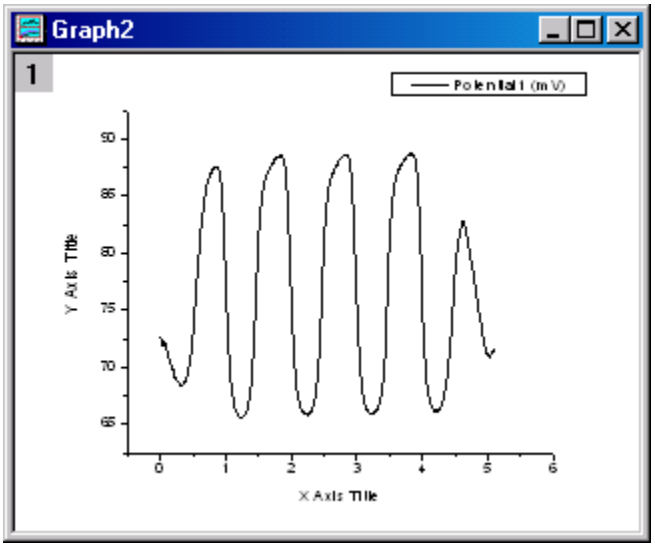
Creating Your Own Multiple Layer Graph Template

Obviously, Origin could not contain pre-defined templates for all the different ways in which layers can be arranged. However, you can create your own custom multiple layer graph and save it as a template for future use.

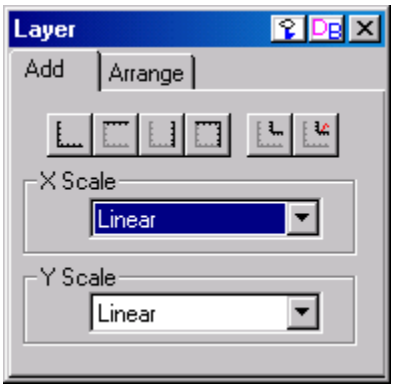
This segment of the tutorial describes a method for creating a multiple layer graph, using a single layer graph as a starting point. This information is provided to familiarize you with important layer concepts. However, keep in mind that you use any of Origin's multiple layer graph templates as a starting point for your custom multiple layer graph.

To begin this portion of the tutorial:

15. Click on the **Potential1(Y)** column heading in the Layers worksheet.
16. Click the **Line** graph button  on the 2D Graphs toolbar.



17. From the menu, select **Tools:Layer** to open the Layer tool.



Note: Make a mental note as to where to find this tool. As the tool's tab names imply, it is very useful for adding and arranging layers in multiple layer graphs. This tool is available only when a graph window is active.

18. On the **Add** tab, click the **Linked Right Y** button  . This adds a second layer to the graph.

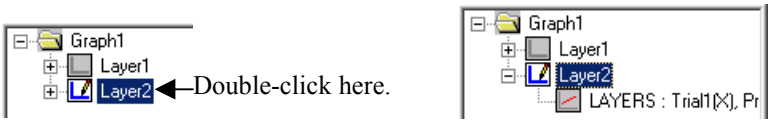
By default, the X axis of this new layer (Layer 2) is linked to the X axis of layer 1. This means that if you change the X axis scale in *layer 1*, the X axis scale in *layer 2* will change simultaneously.

19. Double-click on the layer 2 icon in the upper-left corner of the graph window.

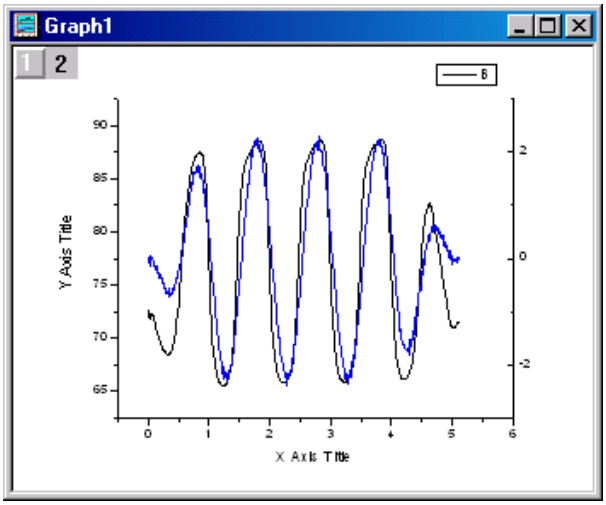
This opens the **Layer** dialog box. What you see to the left side of this dialog is list of all Y datasets in the current project. Note that *no* datasets are currently listed in the **Layer Contents** list box. We have added a second layer to our graph, but we have not added any data to the layer.

20. Select layers_pressure1 in the Available Data list box.
21. Click the 

button to add the layers_pressure1 data set to the **Layer Contents** list box.
22. Click **Layer Properties**. The **Plot Details** dialog box opens.
23. Double-click on the **Layer2** icon on the left side of the dialog box.



24. Select the **LAYERS:Trial1(X), Pressure1(Y)** data plot icon on the left side of the dialog box. This action opens the **Line** tab.
25. Select **Blue** from the **Color** drop-down list, then click **OK**.
26. Click **OK** to close the Layer 2 dialog box.



Note: If you see only the positive peaks in the layers_pressure1 data set, make certain that layer 2 is active and click the **Rescale** button on the Graph toolbar.

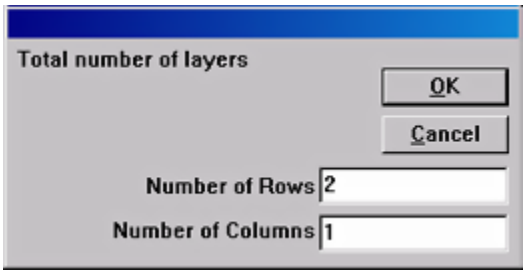
Origin provides many methods to add layers to your graph. In addition to the Layer tool, you can select menu commands from the **Edit** menu. You can also right-click in the gray area outside the graph page (but inside the window) and add layers via the shortcut menu.

Arranging Layers in the Graph Window

In this section, you will add and arrange layers to set up a vertical 2 panel graph with left and right Y axes.

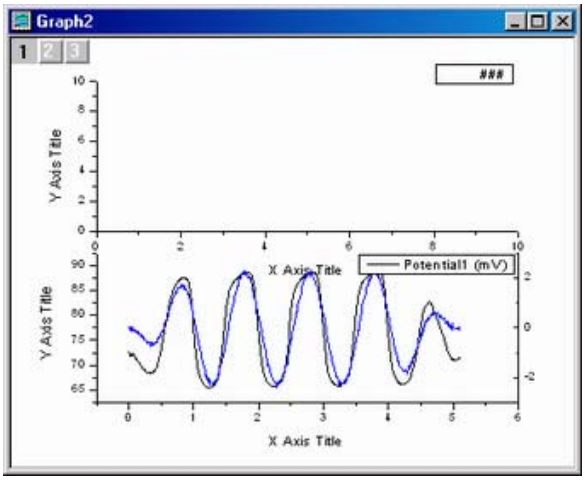
To add and arrange layers in the graph:

27. Select Edit:Add & Arrange Layers.
28. In the **Total Number of Layers** dialog box, type **2** in the **Number of Rows** text box.



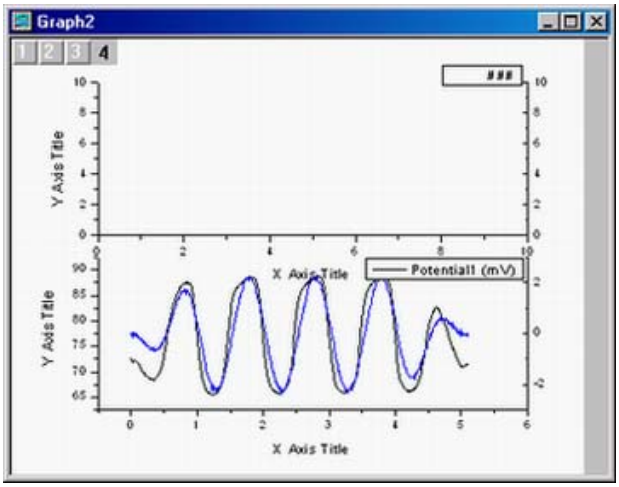
29. Click **OK**. Origin asks for permission to create 1 more layer.

30. Click **Yes**.
31. Click **OK** in the **Spacing** dialog box to accept the default settings.



To add the right Y controlling axis to the top layer:

32. With the layer 3 icon selected, right-click in the gray area inside of the graph window (but outside of the page) and select **New Layer (Axes):(Linked):Right Y** from the shortcut menu.



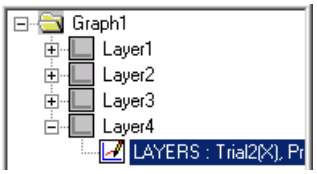
Adding Data to the New Layers

To add the data to layers 3 and 4, you will use the **Layer** dialog box in the same way you added the data to layer 2.

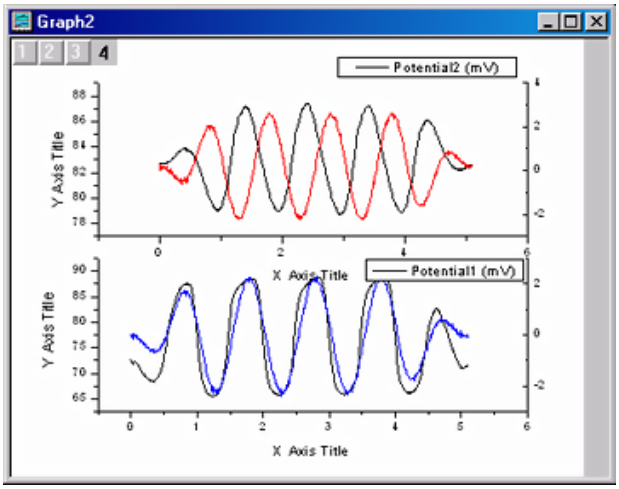
To add data to the new layers:

33. Double-click on the layer 3 icon. The **Layer 3** dialog box opens.
34. In the **Layer 3** dialog box, select **layers_potential2** from the **Available Data** list box, and click  to add the data set to the **Layer Contents** list box.
35. Click **OK**.
36. Double-click on the layer 4 icon.

37. In the **Layer 4** dialog box, select **layers_pressure2** from the **Available Data** list box, and click => to add the data set to the **Layer Contents** list box.
38. Click Layer Properties to open the Plot Details dialog box.
39. Double-click on the Layer4 icon on the left side of the dialog box, then click on the **LAYERS:Trial2(X), Pressure2(Y)** data plot icon.



40. Select **Red** from the **Color** drop-down list.
41. Click **OK** to exit the **Plot Details** dialog box.
42. Click **OK** in the **Layer 4** dialog box.

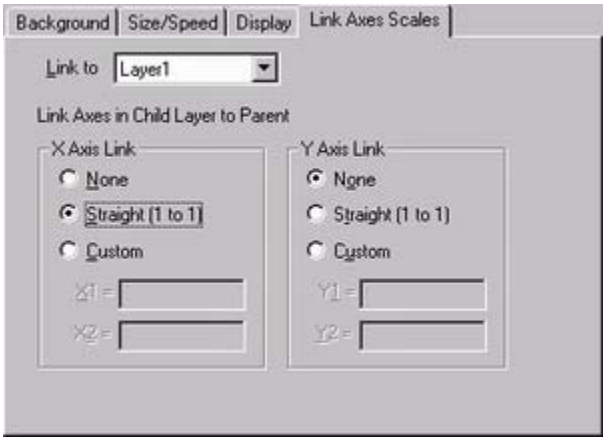


Linking Axes

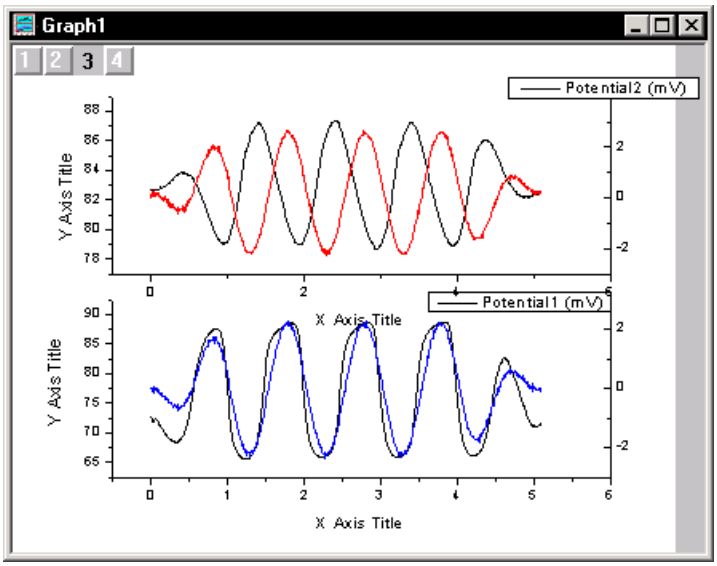
You can link the axes of multi-layer graphs so that when you change the axis scale in the parent layer, the child layer's axis will change accordingly. Axes can be linked dimensionally, by axis scale values, or both.

To Link the X Axes:

43. Double-click on the layer 3 icon to open the **Layer 3** dialog box.
44. Click Layer Properties.
45. Select the Link Axes Scales tab.
46. Select **Layer 1** from the **Link To** drop-down list.
47. In the **X Axis Link** group, select the **Straight (1 to 1)** radio button.



48. Click **OK** to close the **Plot Details** dialog box.
49. Click **OK** in the **Layer 3** dialog box.



You can test the axis link by double-clicking on the bottom X axis (this is Layer 1) and changing the **From** or **To** values on the **Scale** tab. After clicking **OK**, the top X axis will reflect your changes.

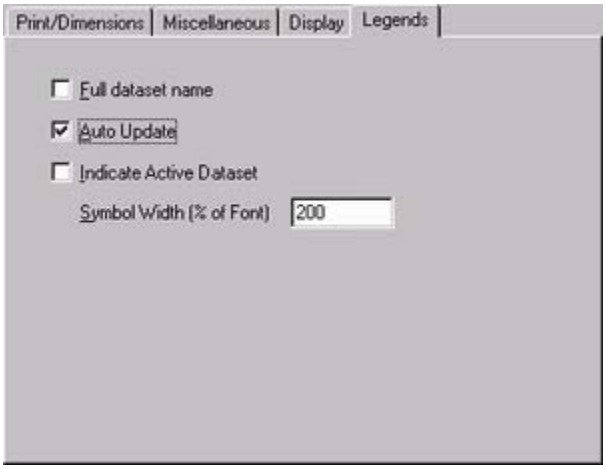
Note 1: Though you only actively linked layer 3 to layer 1, all four layers of this graph are actually linked, directly or indirectly, to layer 1. Recall that when you added layer 2, you added a **linked right Y** axis. Recall also that when you added layer 4, you added a **linked right Y** axis. Thus, layer 2 is linked to layer 1, layer 4 is linked to layer 3, and layer 3 is linked to layer 1.

Note 2: Any change to the physical dimensions or scale values of linked layers should be done to the *parent* layer (this is layer 1 in the above example). Changing dimensions or scale values of a child layer will not effect changes in the parent layer, and the linking criteria will not be respected.

Customizing the Legend

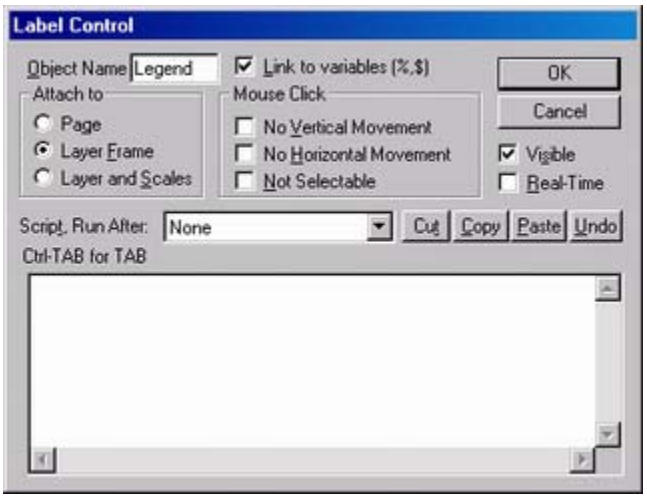
Origin automatically displays a legend when you create a new graph. Furthermore, legends are layer-specific, so that if your graph has multiple layers, Origin displays a legend for each layer. If you add additional data to a layer, the legend will not update to include the new data unless you have selected the

Auto Update check box on the **Legends** tab at the page level in the **Plot Details** dialog box (**Format:Page**).

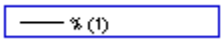


You can update the legend for the *active layer* at any time by clicking the **New Legend** button  on the **Graph** toolbar. If there is currently no legend in the layer, Origin creates one.

A legend is really a specialized text label. Origin names the text label **Legend** to identify it. This name is stored in the legend's **Label Control** dialog box. You can open this dialog box by selecting the legend and then selecting **Format:Label Control**. If you should rename the legend, it retain it's current information but it will no longer update when you click **New Legend** or when you add data to the layer.



A legend text label also uses special formatting to display the data set names and the data plot type icon (such as a line). If you double-click on the legend in the top layer of the current example, the in-place editing mode will display as shown in the following figure.



The %(1) is called substitution notation. This notation instructs Origin to display the data set name for the *first* data set in this layer.

When you are performing an in-place edit of any label, you can use substitution notation. For example, you can display the value of a specific data point in a data plot using the following substitution notation:

%(*WorksheetName*, *ColumnNumber*, *RowNumber*)

For example, if a worksheet is named Data1, and you have plotted columns A(X) and B(Y), you can display the value of the data point in column 2, row 3 using:

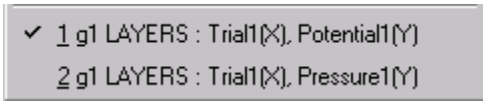
%(data1, 2, 3)

To enable use of this substitution notation, you must select the **Link to Variables** (%,$) check box in the **Label Control** dialog box (see the top of the dialog box, in the above figure).

In addition to substitution notation, Origin uses a special escape sequence to display the data plot icon in the legend. This special escape sequence is:

\L(*DataListPosition*)

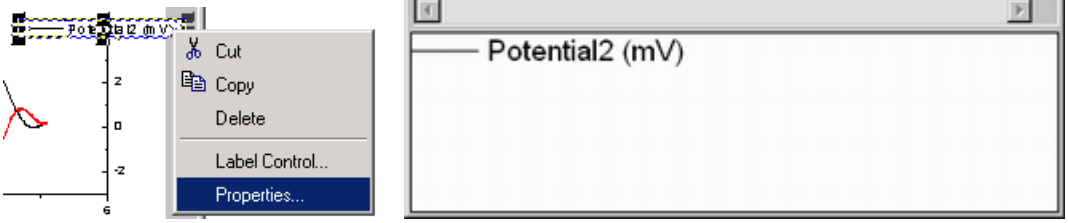
Where *DataListPosition* is the data plot's position in the data list at the bottom of the **Data** menu.



If you double-click on a legend to enter the in-place editing mode, you will see the data plot icon, not the \L( ) escape sequence. This is because Origin does not support using escape sequences in the in-place editing mode. To use escape sequences, you must edit the label in the **Text Control** dialog box (see the following instructions).

To customize the legend:

50. In the bottom layer, click on the text portion of the legend reading Potential1 (mV), then press DELETE.
51. In the top layer, right-click on the text portion of the legend reading Potential2 (mV) and select **Properties** from the shortcut menu. This opens the Text Control dialog box.



52. In the Text Control dialog box, type the following text, overwriting \L(1) %(1):

\L(1.1) Potential1

\L(2.1) Pressure1

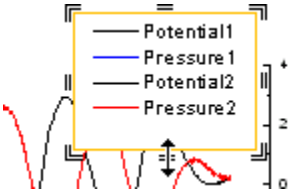
\L(3.1) Potential2

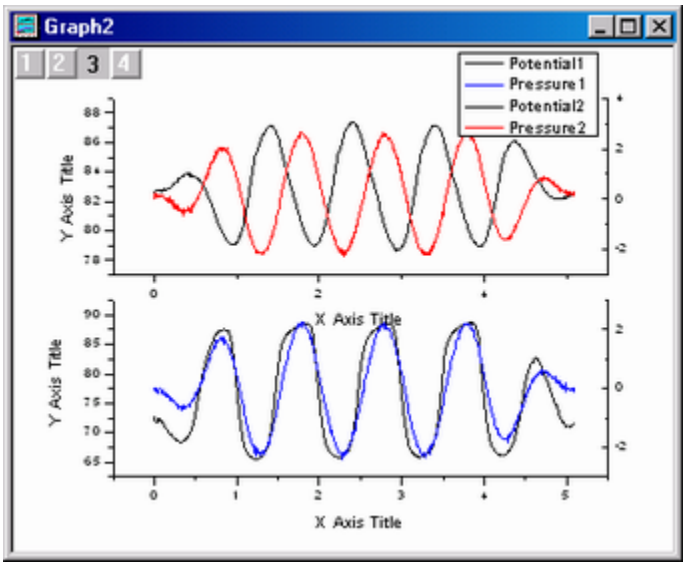
\L(4.1) Pressure2

The preview box on the bottom of the dialog box shows what will display in the legend.

53. Click **OK**.
54. If necessary, drag the legend to a new location to minimize overlapping of graphic elements.

Note: The legend border may be somewhat larger than is needed. To resize the border/background without resizing the legend text, click once on the legend border to activate the "resize border only" handles. When a mouse-over produces a double-headed arrow, use your mouse to resize the legend border. The text point size will not be affected.





The legend now displays the data plot type representations from all the layers in the graph. To prevent Origin from overwriting your custom legend (for example, if you should accidentally click the New Legend button on the Graph toolbar) you should rename the legend.

To rename the legend:

55. Right-click on the legend text.
56. Select **Label Control** from the shortcut menu that opens.
57. Type Custom Legend in the Object Name text box.
58. Click **OK**.

Saving the Graph as a Template

Template files retain information concerning how data should be displayed, but templates do not save any actual data. If you save the current graph window as a template, then the next time you need to create a similar graph, you can select your worksheet data and then select your custom graph template. Your custom template is recalled by clicking the **Template** button on the 2D Graphs toolbar or by selecting **Plot:Template Library**.

To save your graph as a template:

59. Right-click on the graph window title bar.
60. Select **Save Template As** from the shortcut menu.
61. Type **Multilayer** in the **File Name** text box.
62. In the Template Category combo box, type My Templates.
63. Click **Save**.

To test your template, you can make the Layers worksheet active, select all worksheet columns, then select **Plot:Template Library**. Select the **My Templates** category and then select your **Multilayer** template. Click **Plot** to plot your data.

This concludes the **Multiple Layer Graphs** tutorial.

Tutorial 6, 3D Surface and Contour Graphs

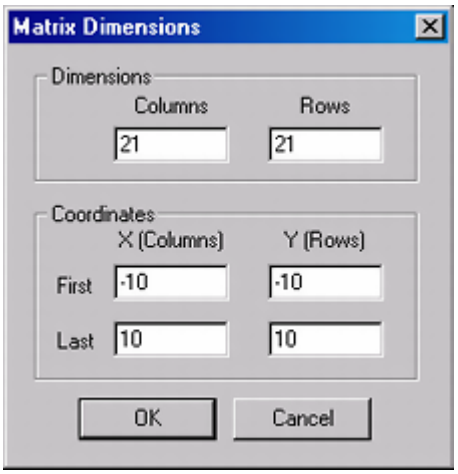
There are two primary data structures in Origin: worksheets and matrices. Data stored in worksheets can be used to create any 2D graph and some 3D graphs, but in order to create a 3D surface graph or a 3D contour graph you must have your data stored in a matrix. Origin provides methods for converting worksheets to matrices, and for converting matrices to worksheets.

An Introduction to Origin Matrices

Matrices have numbered columns which are mapped to linearly spaced X values and numbered rows which are mapped to linearly spaced Y values. You can view the X and Y values in the matrix column and row headings by selecting **View:Show X/Y**. Each cell value in a matrix represents a Z value which is located in the XY plane by the cell's X value (determined by its column) and by the cell's Y value (determined by its row).

To learn more about the XY mapping in a matrix:

1. Click the **New Matrix** button  on the Standard toolbar to create an empty matrix.
2. Select **Matrix:Set Dimensions** to specify its dimensions and coordinates.
3. In the **Dimensions** group, type **21** in the **Columns** text box and **21** in the **Rows** text box.
4. In the **Coordinates** group, type **-10** in both the **First X (Columns)** text box and the **First Y (Rows)** text box.
5. In the **Coordinates** group, type **10** in both the **Last X (Columns**) text box and the **Last Y (Rows)** text box.



6. Click **OK**. Use the horizontal and vertical scroll bars on the matrix window to verify that you have created a matrix with 21 columns and 21 rows.
7. From the Origin menu, select **View:Show X/Y**.
8. Use the horizontal and vertical scroll bars on the matrix window to verify that column (X) and row (Y) headings range from -10 to 10.

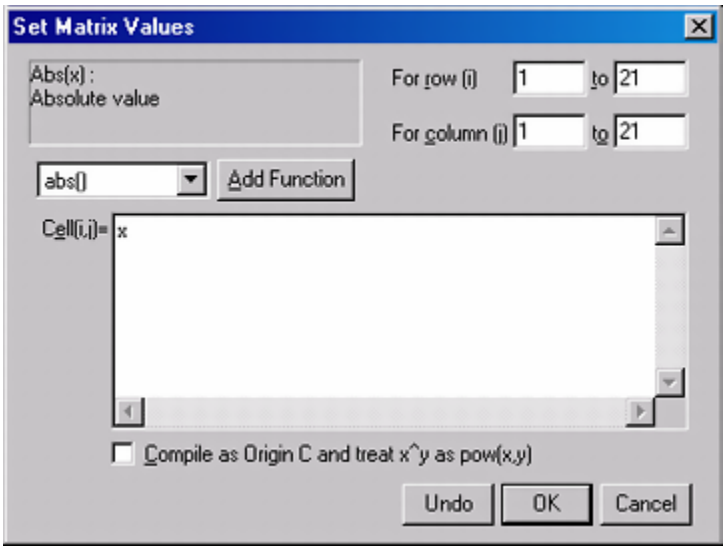
You can see that there are two ways to view the matrix column and row headings – by column or row number (**View:Show Column/Row**)or by the X and Y axis values (**View:Show X/Y**). When we create our 3D plot from this matrix window, you will see that both X and Y axis values range from -10 to 10, so it

should be clear that the values that you specify in the Coordinates group of the Matrix Dimensions dialog will determine your 3D graph X and Y scale values.

9. From the Origin menu, select **Matrix:Set Values**. This opens the **Set Matrix Values** dialog.

We can use this dialog to fill our empty matrix with Z values. You might, for instance, want to plot a mathematical function in 3D. To demonstrate:

10. Type **x** in the **Cell(i,j)=** text box and click **OK**. Each cell in each column is filled with a number equal to the X value of that column.



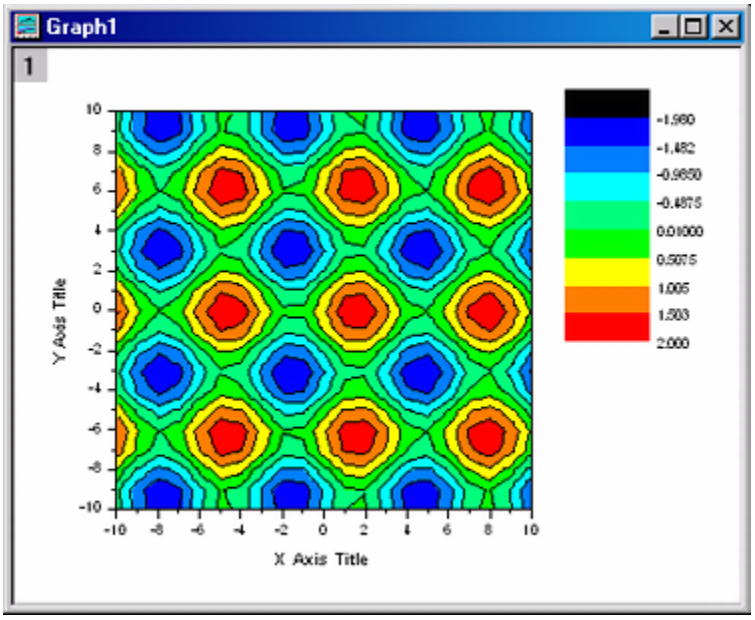
11. Re-select **Matrix:Set Values** and this time type **y** (replacing x) in the **Cell(i,j)=** text box. Click **OK**. Each cell in each row is filled with a number equal to the Y value of that row.

12. Once more, select **Matrix:Set Values** and type the equation **cos(x)+sin(y)** in the **Cell(i,j)=** text box. Click **OK**. The matrix is filled with numbers.

The value in each cell, *z*, is determined by the relationship cos(*x*)+sin(*y*); where *x*=x coordinate value and *y*=y coordinate value. There is no particular significance to our made-up function; we only want to point out that when you specify x or y in a function in the Set Matrix Values dialog, the x and y coordinate values will be used to calculate the z value.

To create a 3D Contour plot from a matrix of values:

13. Select **Plot:Contour Plot:Contour-Color Fill** to create a contour graph of the matrix data. Note the X and Y scales (-10 to 10) on the resulting graph.



Converting a Worksheet to a Matrix

In many, if not most, cases, your data will be stored not as a worksheet or spreadsheet of z values but rather as a worksheet of x,y,z triplets. When this is the case, you must convert your worksheet to a matrix of z value before you can create a 3D surface or contour plot. In this section you will learn how to convert an XYZ worksheet to a matrix so that it can be plotted as a 3D surface graph.

The data for this lesson is provided in an ASCII file.

To Import the ASCII File:

14. Click **New Project** on the Standard toolbar.
15. Click **Import ASCII** on the Standard toolbar.
16. In the Origin \Tutorial folder, select Tutorial_5.DAT from the list of files.
17. Click **Open**.

Tutorial5

	A[X]	B[Y]	C[Y]
1	2.5536	43.65282	1.506E-6
2	5.1072	18.96488	2.655E-6
3	7.6608	-0.50993	2.705E-6
4	10.2144	-17.29591	4.291E-7
5	12.768	-40.6861	3.155E-7
6	15.3216	36.13428	1.927E-6
7	17.8752	16.20608	5.202E-6
8	20.4288	-4.96125	2.614E-6
9	22.9824	-22.80867	8.96E-7
10	25.536	-38.39416	5.74E-7

By default, when the file is imported, columns are added to the worksheet as Y columns. To convert the worksheet to a matrix it must be in an XYZ format. Recall that we can change a column's plot designation in a couple of ways:

- Double-click on a worksheet column heading to open the Worksheet Column Format dialog box.
- Right-click on a column heading and choose Set As (X, Y, Z, etc.) from the shortcut menu.

To Change the Column Designation:

18. Right-click on the C(Y) column heading.
19. Select **Set As:Z** from the shortcut menu. Column C is now designated as a Z column.

Selecting the Type of Conversion

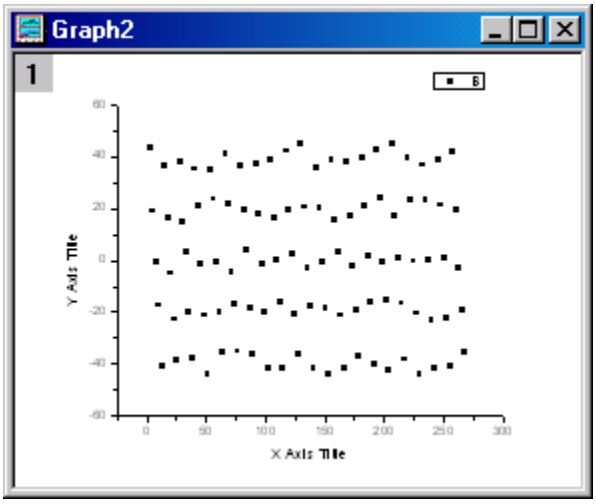
Origin provides several methods for converting worksheets to matrices, including direct, expand columns, 2D binning, regular XYZ and random XYZ conversions. The method that you choose will depend on the nature of your worksheet data.

We will give a brief run-down of the matrix conversion methods available to you, but we also encourage you to study the information in Worksheet to Matrix.OPJ in the \Samples\Analysis\Worksheet to Matrix folder in your Origin software directory.

- **Direct Conversion**: A common format for XYZ data is to have X data values in the left-most column, Y data values in the first row, and Z values in columns 2 to *N* and rows 2 to *M*. If your worksheet (or Excel spreadsheet) data are organized in this way, you should probably use Direct Conversion.
- **Expand Columns**: There may be cases where a screen editor which is used to create an ASCII file has an upper limit for the number of characters allowed on a single line. This upper limit may even be less than the number of columns required in the matrix. To compensate for this, one row of matrix data can be stored in multiple rows of an ASCII file. After this type of data is imported in a worksheet, the Expand Columns method will copy the number of worksheet rows that you specify into the first row of a matrix. Origin repeats this procedure to convert all rows in the worksheet.
- **2D Binning**: This method bins the XY data (creates a frequency count of data points falling within a given XY range) and stores the bin counts as Z values in a matrix.
- **Regular XYZ**: In order for XYZ worksheet data to be considered Regular (a) each X value must have the same number of Y values and each Y value must have the same number of X values and (b) both the X and the Y data values must be equally
- **Sparse XYZ**: This method converts "regular" XYZ data to a matrix by assuming missing values for missing X,Y data pairs.
- **Random XYZ**: If your worksheet data are not of the previously mentioned types, then it can be classified as random data. The procedure for converting random XYZ data to a matrix is called gridding. Origin offers a number of gridding methods, including the method of Renka and Cline and a modification of Shepard's method. Both these methods are provided by the NAG® C Library eo1 Interpolation.

Most likely, you will use either the **Regular XYZ** or **Random XYZ** worksheet to matrix conversion methods. There is a simple test that you can use to determine which method is best suited to your data. This involves selecting only the X and Y columns of your XYZ worksheet and plotting the XY data as a scatter plot. If the resulting plot resembles an "orchard in plan view" (each scatter point is a "tree"), you can conclude that the data are *Regular XYZ* data. If the resulting plot displays no such regularity, it is *Random XYZ* data.

20. Activate the Tutorial5 worksheet, select the X and Y columns of data, and click the **Scatter Graph**  button. The resulting plot looks like this:



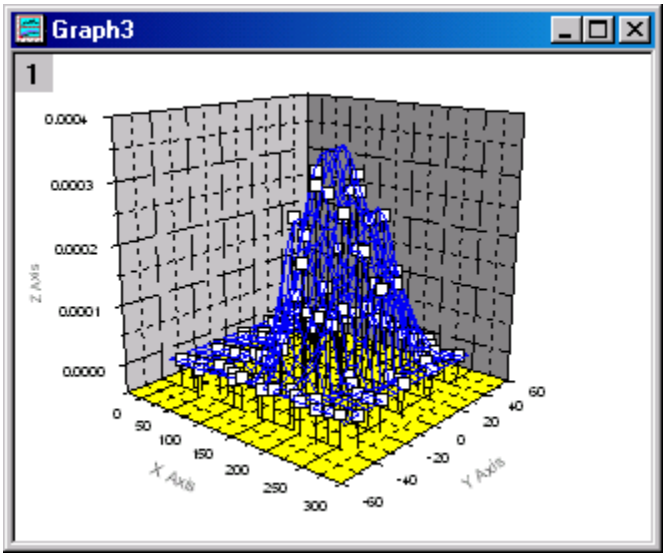
Clearly, our data are not regularly spaced in the X and Y dimensions, so we will convert our worksheet to a matrix using one of Origin's Random XYZ gridding methods.

To convert the worksheet to a matrix:

21. In the Tutorial5 worksheet, select the C(Z) column heading.
22. Select **Edit:Convert to Matrix:Random XYZ**. This opens the Random XYZ Gridding dialog box.

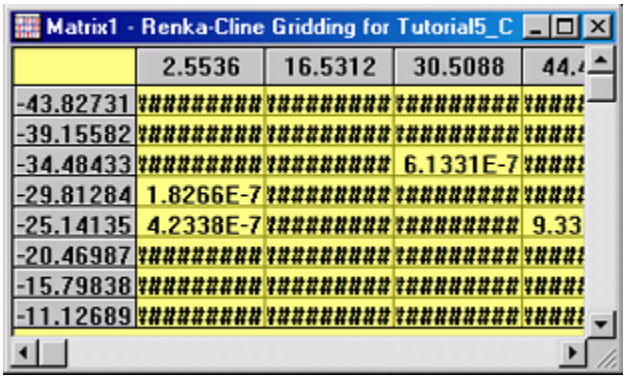
23. Select **Renka-Cline** from the **Select Gridding Method** drop-down list if it is not already selected.
24. Select the **Show Plot** check box if it is not already selected.
25. Click **Apply**. A graph is created that shows the raw data as an XYZ scatter plot (the Show Plot check box) and the gridded data as a 3D Wire Frame plot.

The superimposition of the raw data on the gridded data allows you to quickly assess a gridding method's effectiveness. Obviously, you don't want your gridded data to depart radically from your raw data.



You can change the gridding method – each time clicking the Apply button – to compare the graph output for the various methods and then choose the method which produces a surface that most closely coincides with your raw data. For our example, the Renka-Cline method is the best choice.

26. Click **OK** to close the dialog box.
27. Activate the matrix window that contains the Renka-Cline gridding data. (Note that if Origin cannot display an entire cell value, it displays # characters in the cell. How to increase column width of the matrix so that the entire cell value is visible).

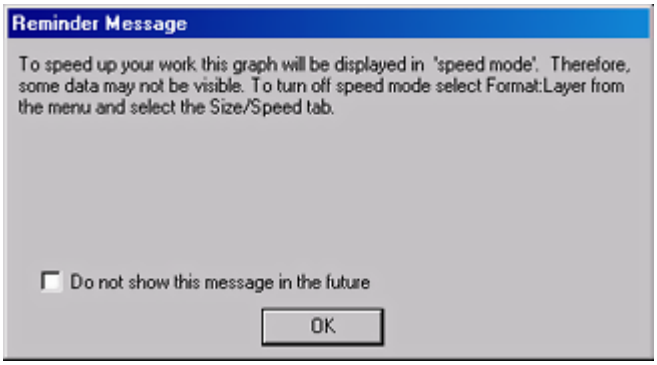


Graphing and Speed Mode

Now that you have your data in a matrix you can create any of Origin's contour or 3D surface graphs. For this tutorial you will create a 3D color mapped surface graph.

To create a 3D color mapped surface graph:

28. With the Renka-Cline gridding matrix active, select **Plot:3D Color Map Surface**. A Reminder Message reports that the graph will be displayed using *Speed Mode*.



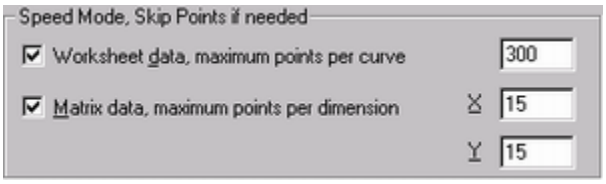
It is worth taking a moment to discuss "speed mode" as it is a useful but sometimes confusing option for *both* 2D and 3D graphs. Speed mode can be very useful for increasing the screen redraw speed of your graph by displaying a reduced number of data points. While the effect is not noticeable for plots created from smaller matrices, it can be very significant for plots of very large matrices.

At other times, you may find that reducing the number of plotted points is the only way to produce an intelligible data plot; imagine, for instance, trying to create a scatter plot of 1 million data points.

Speed mode reduces the number of plotted by points by systematically weeding out every *n*th data point from your dataset. Speed mode can be invoked at three levels depending on your requirements:

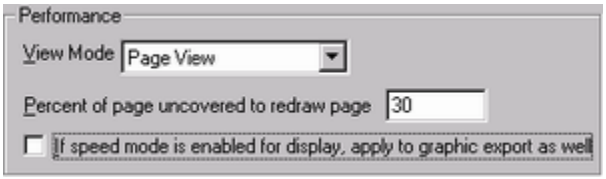
- Video display
- Clipboard Images and Graphic Export files
- Printout

At the video display level, you can control speed mode by activating your graph and selecting **Format:Layer** from the menu. This opens the active layer's Plot Details dialog box. Select the **Size/Speed** tab and edit the **Speed Mode, Skip Points if needed** group.

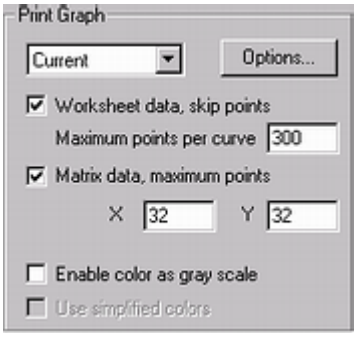


This speed mode setting only effects the display of the graph in Origin. When you *copy, export,* or *print your graph, all the data points will be included*.

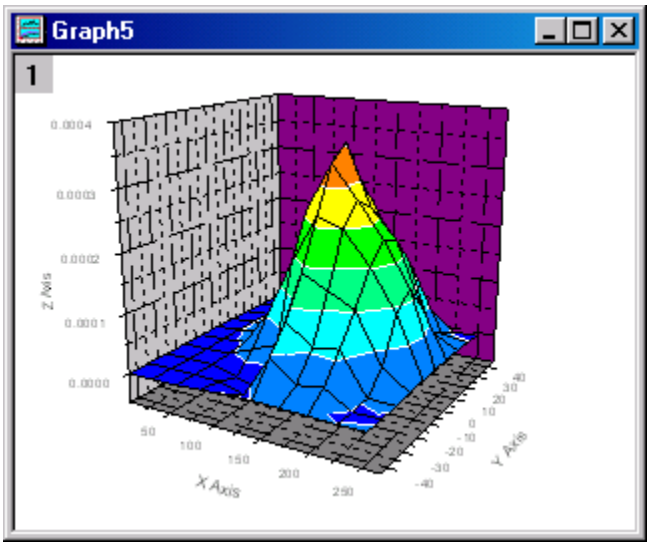
To invoke speed mode for Clipboard (copied) images and graphic export files (BMP, TIF, JPG, etc.), select **Format:Page.** This opens the Plot Details dialog box. Select the **Miscellaneous** tab and, in the **Performance** group, select the **If speed mode is enabled for display…** check box. This will apply your speed mode settings to your copied/exported graph.



Lastly, you can reduce the number of data points in your printouts, using the settings in the **Print Graph** group of the Print dialog box (**File:Print** from the Origin menu).

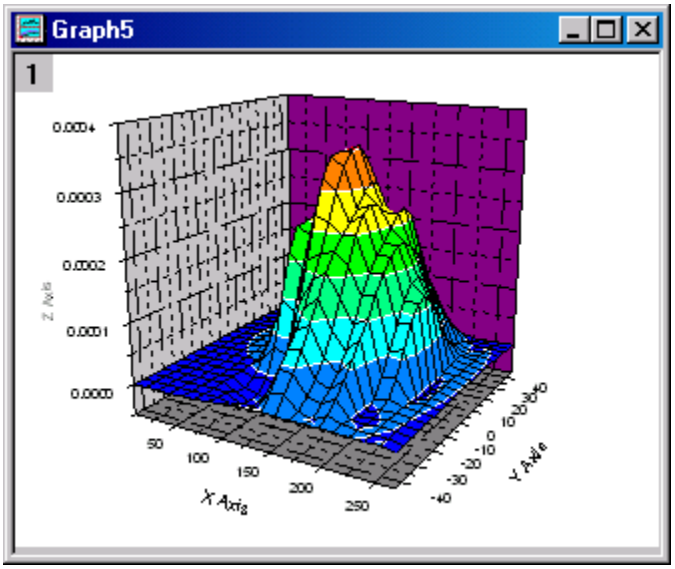


29. Click **OK** to close the speed mode reminder message. The matrix data is plotted as a color map surface graph. The different colors represent different Z-value ranges.



30. From the Origin menu, select **Format:Layer**. This opens the Plot Details dialog.
31. Select the **Size/Speed** tab.
32. In the Speed Mode, Skip Points if needed group, clear the Matrix data, maximum points per dimension check box and click OK.

Note that the appearance of your graph has changed somewhat. The default speed mode settings limited the portion of the matrix that plotted to 15 rows and 15 columns, but by clearing this check box, we are now plotting the full20 rows by 20 columns of data.



Customizing the Graph

Origin gives you full control over the color mapping applied to the surface data plot. You can customize the color map by changing the settings on the **Color Map** tab of the Plot Details dialog box.

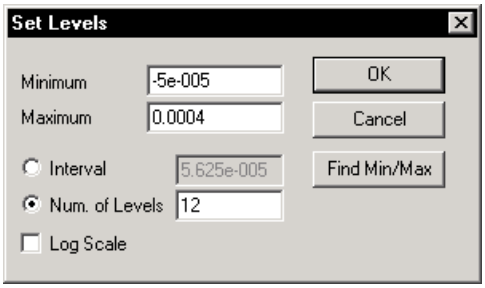
The Color Map tab displays the current color map; with different colors representing a particular range of z values from z_{MIN} to z_{MAX}. To edit an individual level or color, click on the value or color in the Level or Fill column. To edit the entire range of levels or colors, click on the Level or Fill column heading. To edit a range of levels, SHIFT + click on the desired values to select a range, then click on the Level or Fill column heading.

To change the number of levels in the color map:

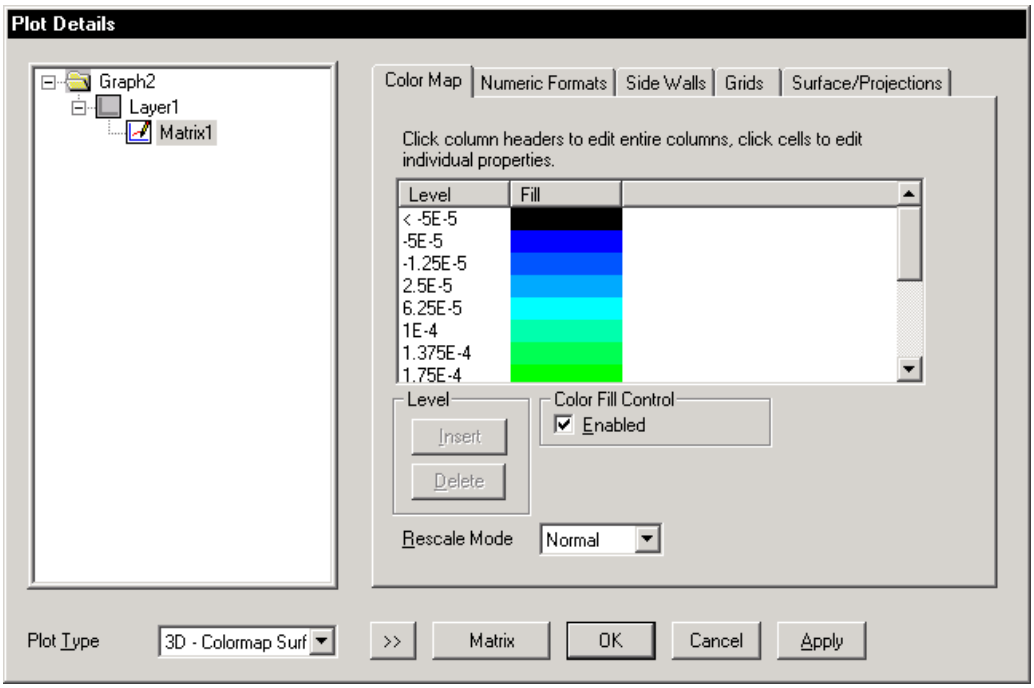
33. Right-click on your 3D Color Map Surface plot (take care not to click on the contour lines or on the side walls) and select **Plot Details** from the bottom of the shortcut menu.

Note: If you have a problem getting to Plot Details from the shortcut menu, you can choose **Format:Plot** from the Origin menu. Both will open the Plot Details dialog to the Color Map tab.

34. Click on the **Level** column heading to open the **Set Levels** dialog box.
35. Select the **Num. of Levels** radio button, type **12** in the associated text box, and click **OK**.



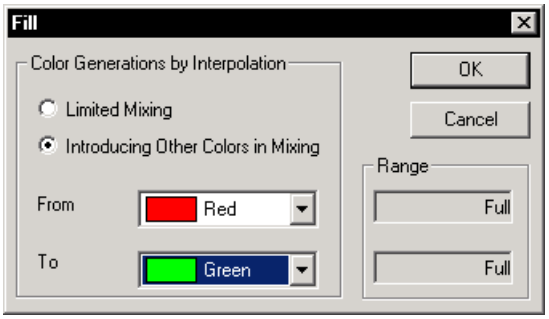
The Color Map tab updates to show twelve levels (plus levels for values above and below the maximum and minimum levels) and associated colors in the list box.



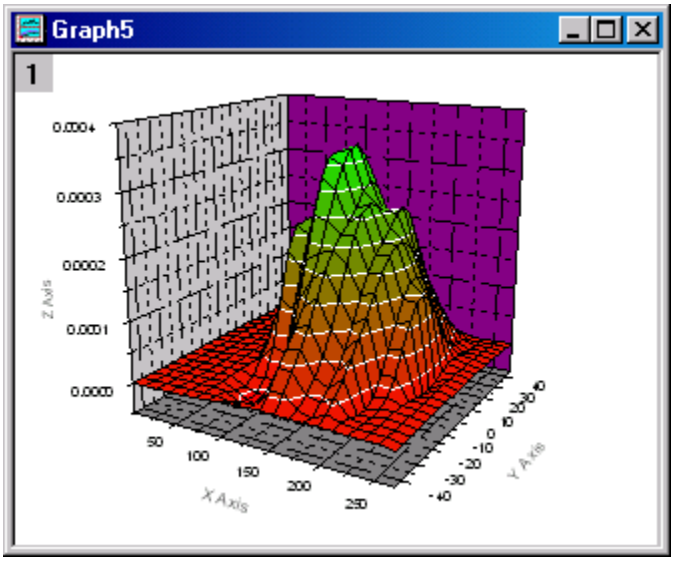
Customizing the Color Map

To customize the color fill characteristics:

36. Click on the **Fill** column heading to open the **Fill** dialog box.
37. From the **From** drop-down list, select **Red**
38. From the **To** drop-down list, select **Green**.



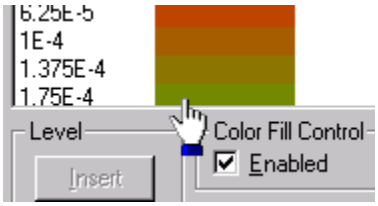
39. Click **OK**.
40. In the Plot Details dialog box, click **OK**.



In addition to editing color ranges, you can edit individual colors. This is especially useful if you have an important section of your data that you want to highlight or make transparent.

To edit an individual color:

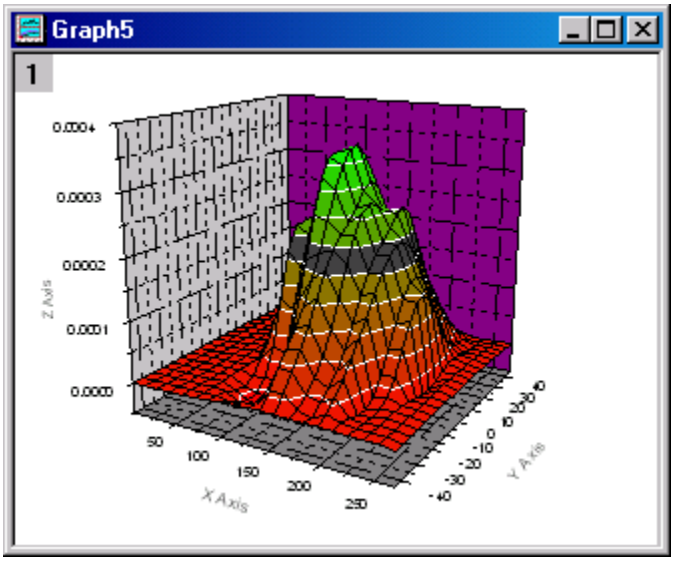
41. Right-click on the color map surface plot.
42. Select **Plot Details** from the shortcut menu.
43. On the **Color Map** tab, click on the color associated with 1.75E -4.



This opens the Fill dialog. Note that this controls the fill color only for Level 1.75E-4.

44. Select **Dark Gray** from the **Fill Color** drop-down list and click **OK**.
45. In the **Plot Details** dialog box, Click **OK**.

The data plot redraws with the 1.75E-4 level in dark gray.



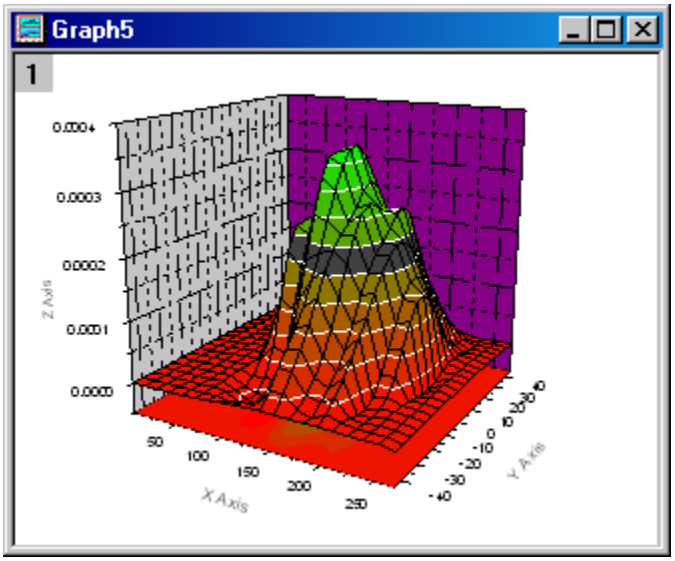
Adding Contours to the Color Map Surface Graph

To further enhance your graph, you can display contour lines or project your color map onto the top or bottom plane of your surface graph. This is done on the Surface / Projections tab of the Plot Details dialog box.

To project your color map onto the bottom plane of the 3D Color Map Surface graph:

46. Right-click on the surface graph.
47. From the shortcut menu, select **Plot Details**.
48. Select the Surface / Projections tab.
49. Under **Bottom Contour**, select the **Fill Color** check box

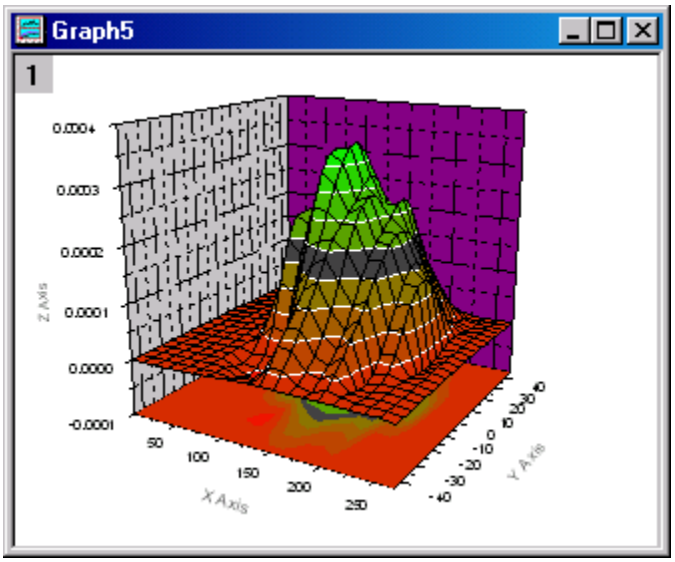
50. Click **OK**.



With the current Z axis scale range and the current view angle, the surface plot substantially overlaps the bottom contour, blocking it from view. To make more of the contour visible, change the Z axis scale to begin from a lower value.

To Change the Z Axis Scale:

51. Select **Format:Axes:Z Axis** to open the Z Axis dialog box.
52. In the **From** text box, type **-1E-4**.
53. Click **OK**.



The Z axis now displays a greater range below the surface graph. This decreases the amount of overlap of the surface plot and the projection, providing for a better visual presentation.

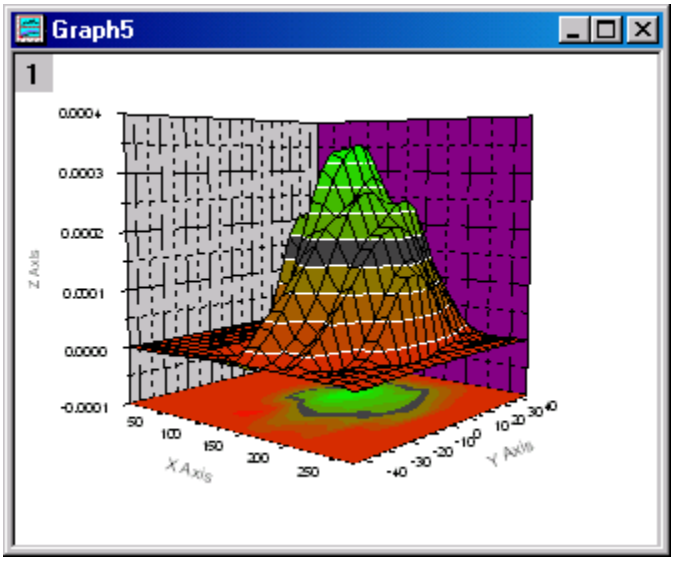
Changing the Perspective of the Graph

When you create a 3D graph, the 3D Rotation toolbar automatically opens. This toolbar provides buttons for controlling the perspective of the 3D graph. By rotating the graph you can further reduce the overlap of the surface plot with the contour, providing better perspective for viewing the graph.

To rotate the graph:

54. On the **3D Rotation** toolbar, click **Tilt Up**  .

55. On the **3D Rotation** toolbar, click the **Rotate Clockwise** button  .



This new perspective eliminates much of the overlap between the surface plot and the projection, and provides better visibility of the dark gray section of the surface plot.

This concludes the **3D Surface and Contour Graphs** tutorial.

Tutorial 7, Nonlinear Curve Fitting

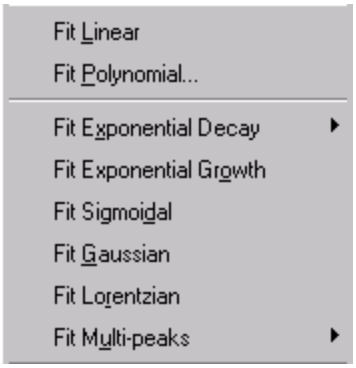
Origin offers several methods of fitting functions to your data. These methods vary in speed and complexity so that you may choose the method that best fits your needs.

In this tutorial, you will be introduced to fitting using the menu commands, the tools, the fitting wizard (NLSF Wizard), and the advanced fitting tool (NLSF). You will then use the NLSF to define your own function and fit sample data.

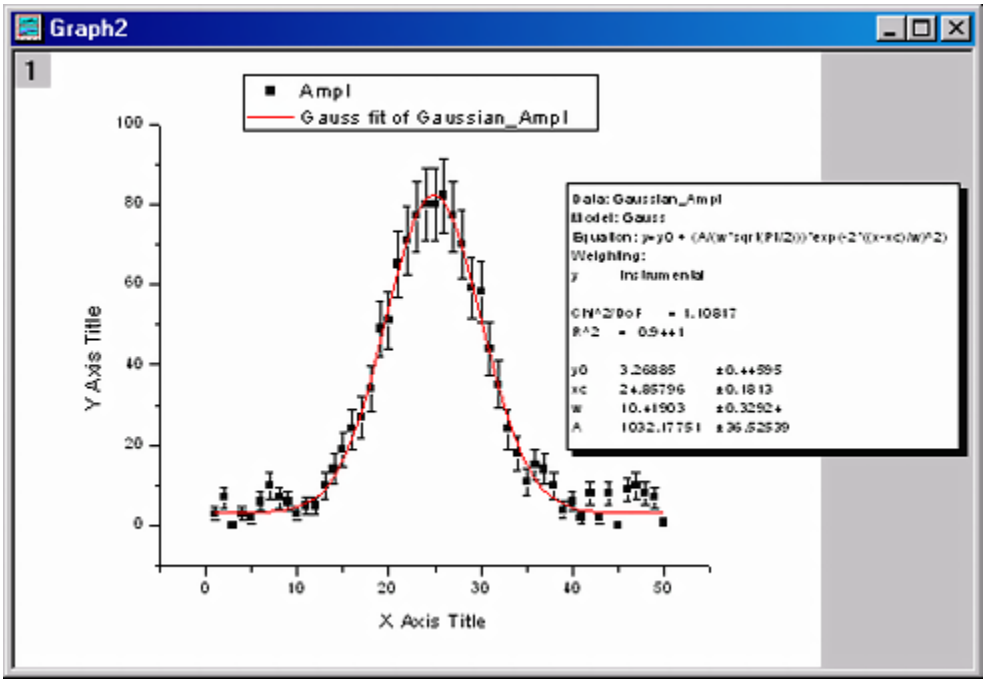
Fitting from the Menu

Origin offers access to several fitting functions directly from the **Analysis** menu. To perform a fit on your data using the menu commands, make sure that the data plot you want to perform the fit on is active, then select the type of fit you want to perform from the **Analysis** menu.

Most of the menu commands require no parameter input from you; the fit is performed automatically. Some may ask you for some parameter information, but will suggest default values based on your data.



After performing the fit, Origin displays the fit curve in the graph window, and types the results to the Results Log.



[7/30/2002 15:04 "/Raw Data/Graph2" (2452485)]
Gauss fit to Gaussian_Ampl:
Weight given by Gaussian_Err error bars.

Chi^2/DoF	1.10817
R^2	0.9441

Area	Center	Width	Offset	Height
1032.2	24.858	10.419	3.2689	79.044

To perform a fit using one of the fitting functions available from the Analysis menu:

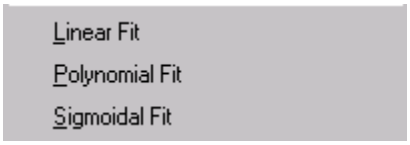
1. On the Standard toolbar, click the **Open** button  .
2. In the \Tutorial folder, open the file FITEXMP1.OPJ. The project contains one worksheet and one graph.
3. Verify that the graph is the active window. From the menu, choose **Analysis:Fit Exponential Decay:First Order**.

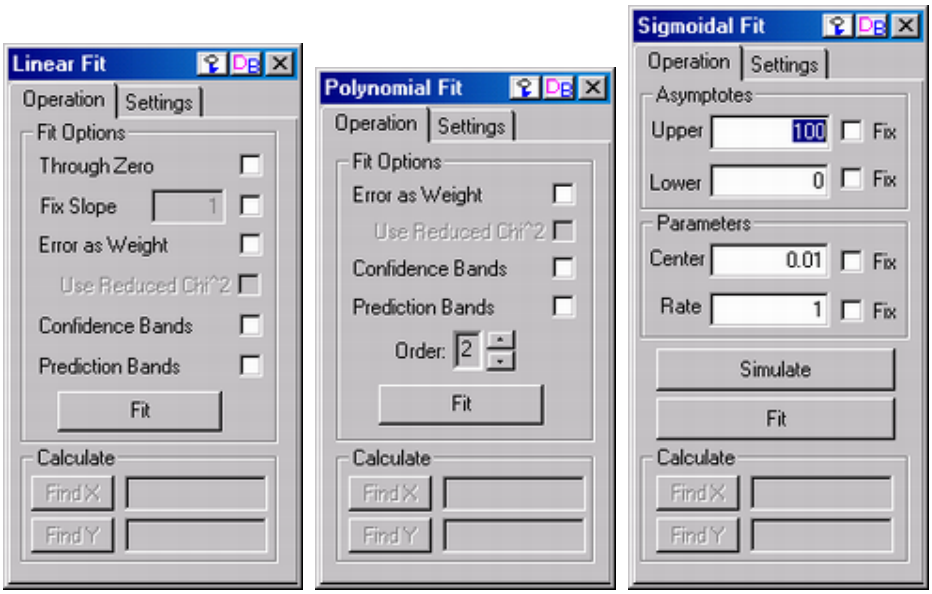
Origin generates a best-fit model for the dataset using the ExpDec1 fitting function. The best-fit curve is plotted as a red line through the data plot.

The Results Log reports data relevant to the fit: the date and time, the graph page, the curve fitting equation, chi-square and r-square values, and the parameter and parameter error values.

Fitting Using the Tools

For those who require a greater degree of control over your curve fit operations than the menu commands allow, Origin provides three fitting tools: the Linear Fit, Polynomial Fit, and Sigmoidal Fit tools. These tools can be opened from the **Tools** menu when either a worksheet or a graph is the active window.

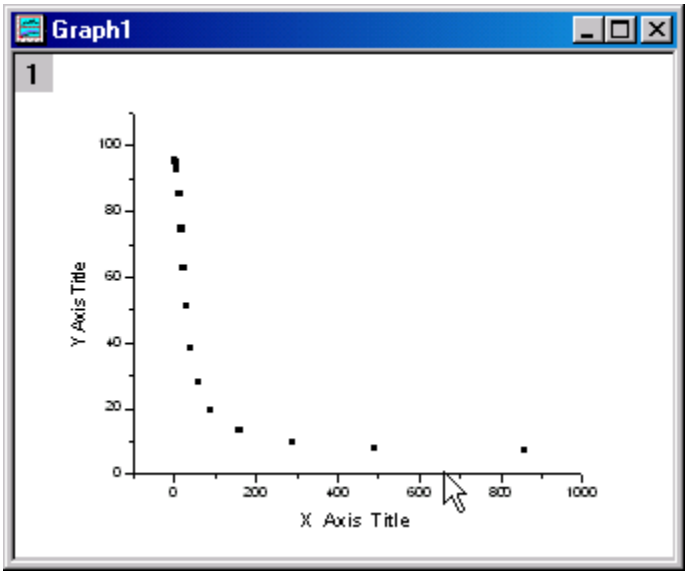




To use the fitting tools, (1) select the data set or data plot you want to fit, (2) open the tool and customize the options on both tabs of the tool, and (3) click the **Fit** button on the Operation tab.

To perform a fit using the Sigmoidal Fit tool:

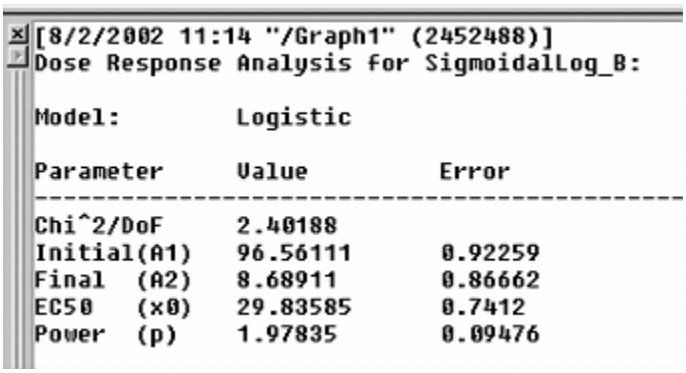
4. On the **Standard** toolbar, Click the **Import ASCII** button .
5. In the \Samples\Data folder, open a file named SigmoidalLog.DAT.
6. In the SigmodalLog worksheet, select the B(Y) column and click the **Scatter** button , or, from the menu, choose **Plot:Scatter**. A scatter graph is created.



7. Place your cursor precisely on the X axis (not on the axis tick labels) and double-click with your mouse. The **X axis** dialog box will open to the **Scale** tab (if you missed with your mouse-clicks and got another tab, just select the Scale tab).
8. From the **Type** drop-down list, choose **Log10** and click **OK**. The X scale changes to a log scale and the plot has taken on a sigmoidal shape.
9. From the menu, select **Tools:Sigmoidal Fit**. The Sigmoidal Fit tool opens.

You will notice that Asymptotes and Parameters groups display some initial values. You can accept the defaults. In most cases, you will not need to adjust these, as Origin will initialize the parameters once you click the **Fit** button.

10. Click the settings tab and notice that the Logged Data Fit Function group has two options for log scaled data: **Boltzmann** and **Dose Response**. We will leave the **Dose Response** function selected.
11. Return to the **Operation** tab and click **Fit**. Origin performs a sigmoidal fit on the data and types the results to the Results Log



```
[8/2/2002 11:14 "/Graph1" (2452488)]
Dose Response Analysis for SigmoidalLog_B:

Model:          Logistic

Parameter       Value           Error
------------------------------------------------
Chi^2/DoF       2.40188
Initial(A1)     96.56111        0.92259
Final   (A2)    8.68911         0.86662
EC50    (x0)    29.83585        0.7412
Power   (p)     1.97835         0.09476
```

Note that the Logistic model was used. This model is commonly used for dose response studies in pharmacology. The formula for this, and other Origin fitting functions, can be found in the .FDF file (a text file) that resides in the \FitFunc folder. The formula for this particular function:

```
y = A2 + (A1-A2)/(1 + (x/x0)^p)
```

We should point out some of the post-fit functionality of the **Sigmoidal Fit** tool. At the bottom of the operation tab is a FindX/Find Y group that allows you to find a Y value from an inputted X value or an X value from an inputted Y value.

The methods differ for FindX and FindY, but the calculation of the X value from an inputted Y is very straightforward. The Y value is calculated using the best fit model that we just calculated using the Sigmoidal Fit tool. We merely input an X value and Origin calculates the Y value using the above equation and parameter values.

To find a Y value based on an inputted X value:

12. Enter **40** in the **FindX** text box.
13. Click **FindY**. Origin calculates a Y value of 40.22945.

You could get out your calculator and see for yourself that our best fit model will return a value of 40.22945 when X=40, but rather than do that, let's digress here for a moment and have a bit of fun.

14. On the Standard toolbar, click the Script Window button  . This opens the Script window.

Origin's Script window has many uses, but one of the simplest applications is to use it to perform calculations. We demonstrate this in one of the later Command Line Tutorials, but let's get our toes wet now and learn a bit more about one of Origin's many capabilities.

15. In the Script window, type:

    ```
    A1=
    ```

    Press **ENTER**.

Origin responds with `96.56111`. Note that this is the value Origin calculated for the parameter `A1`. Origin has temporarily stored our parameter values in memory.

16. Now let's type our Logistic equation into the Script window:

    ```
    y = A2 + (A1-A2)/(1 + (x/x0)^p)
    ```

    Press **ENTER**.

Origin responds by placing a semicolon at the end of the equation.

Now, let's give Origin our X value of 40 and see it return our Y value.

17. Type in the following into the Script window:

    ```
    x=40;
    ```

Press **ENTER**.

Origin responds by placing a semicolon at the end of the line.

18. Now type in:

```
y=
```

Press **ENTER**.

Origin places a semicolon at the end of the line and returns:

```
Y=40.22945
```

This is, of course, our FindY value.

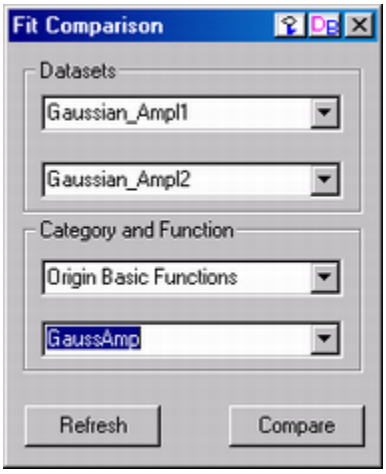
In this example, we used the Origin script window to calculate a Y value given an equation and an X value. The equation parameter values were determined when we performed our sigmoidal fit and they remained in memory because we had not overwritten or changed the value of those parameters.

The point of this little digression was simply to demonstrate that Origin is a very powerful program. We hope that as you begin to work with Origin, that you will come to appreciate the fact that it is much more than a top-notch scientific graphics package. It is also a very powerful analysis package that provides a wide variety of tools for "massaging" and analyzing your data.

Now, let's return to our curve fitting tutorial.

Fitting Comparison

A **Fit Comparison** tool is available to determine if two data sets are representative samples drawn from the same population. This tool compares the data sets by fitting the same function to the data. It then uses an F-test to determine whether the data sets are significantly different from each other. To open the Fit Comparison tool when a worksheet or graph is active, select **Tools:Fit Comparison**.



To compare two datasets using the Fit Comparison tool:

19. Click the **Import ASCII** button  .
20. In the \Samples\Data folder, open a file named Lorentzian.DAT.
21. Select columns B(Y) and C(Y).
22. From the menu, select **Tools:Fit Comparison…**
23. Verify that the **Datasets** group registers **Lorentzian_B** and **Lorentzian_C**.
24. Category should be set to Origin Basic Functions.

25. From the **Function** drop-down list, select **Lorentz**.
26. Click **Compare**.

Origin performs a comparison of the selected datasets and types the result to the Results Log. In this particular case, Origin finds that, at the 0.05 level, the two datasets are statistically different; it is highly unlikely that they represent samples drawn from the same population.

Note: The **Refresh** button should be used to update the **Datasets** or **Category and Function** drop-down lists when, for instance, different datasets are selected for comparison , new datasets are added to the project file, or new functions are defined in the Nonlinear Least Squares Fitter (NLSF).

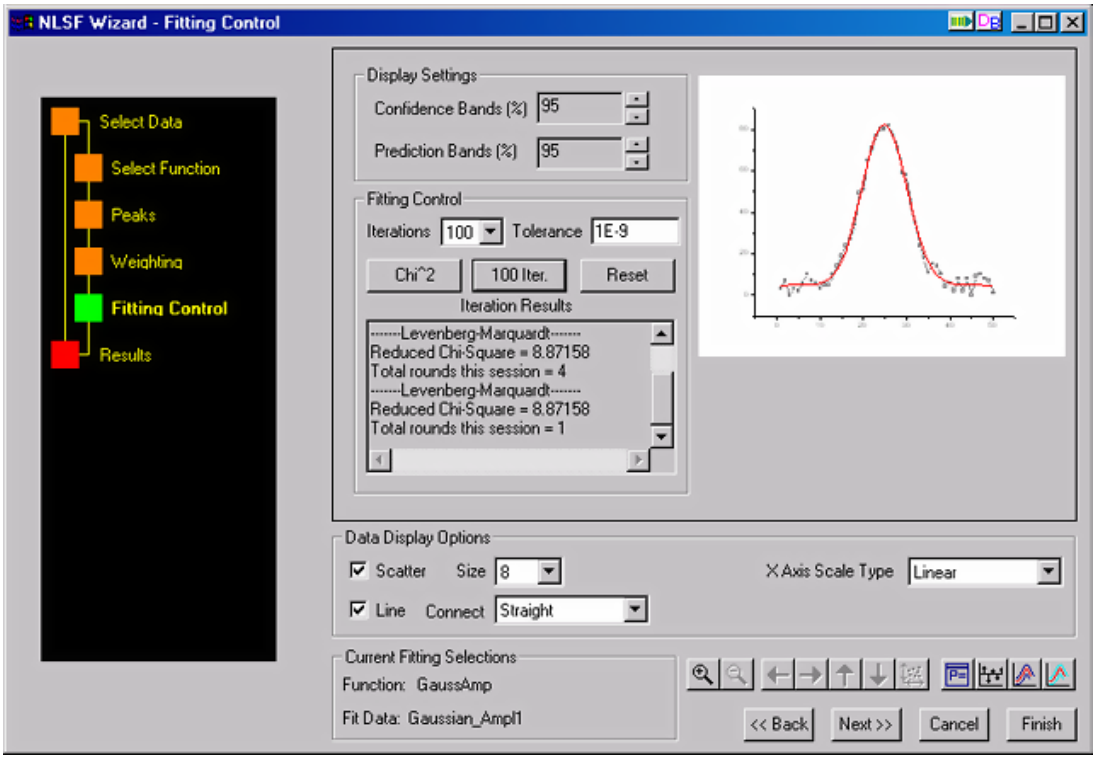
The Fitting Wizard

Origin provides two, progressively more advanced tools for performing nonlinear least squares fitting: the NLSF Wizard and the Advanced Fitting Tool (the NLSF, or Nonlinear Least Squares Fitter).

The wizard – as it's name implies – is quite easy to use but it doesn't allow as much control over the fitting process as the NLSF does. For instance, you cannot define a new fitting function using the Wizard; this must be done through the NLSF (though once defined via the NLSF, user-defined functions become available in the Wizard). In addition, to fit multiple data sets to a function, fit with multiple independent or dependent variables, or run scripts at specified triggers (for example, after a fit is performed), you must use the NLSF.

Yet, the Wizard provides enough control to be an effective and flexible fitting tool. It is highly customizable, allowing you to do such things as hide unneeded pages and save your procedure to a button for repeat analyses using other datasets.

To open the NLSF Wizard when a worksheet or graph is active, select **Analysis:Nonlinear Curve Fit:Fitting Wizard**.



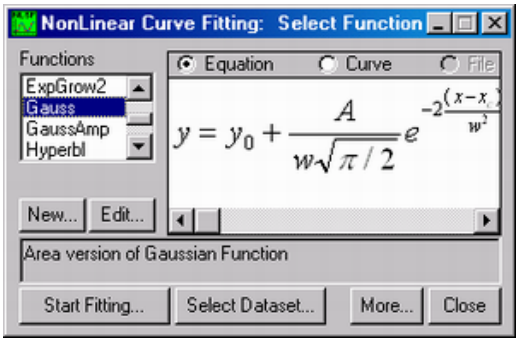
The Advanced Fitting Tool

The advanced fitting tool (NLSF) is Origin's most powerful and complex method of fitting data. To open the NLSF when a worksheet or graph is active, select **Analysis:Nonlinear Curve Fit:Advanced Fitting Tool**.

There are two display modes available for the NLSF: basic and advanced. You can switch between modes by clicking the More button in the basic mode or the Basic Mode button in the advanced mode.

The Basic Mode

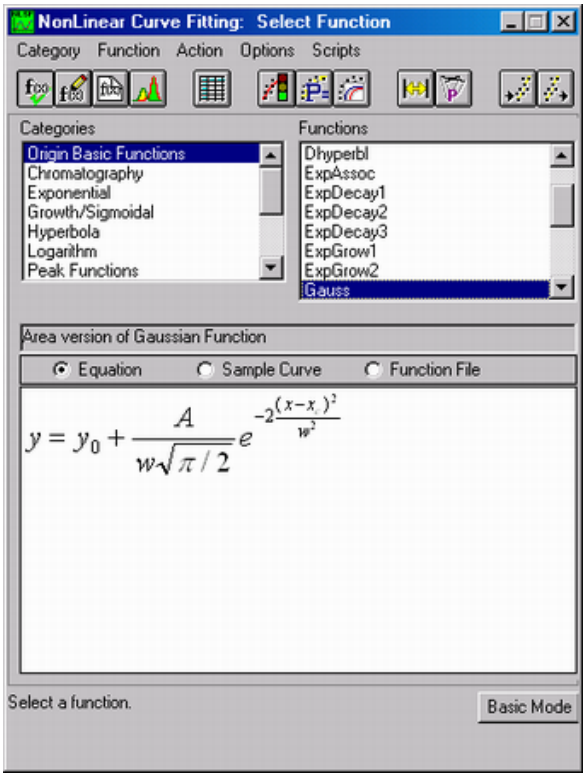
The basic mode of the NLSF provides an abbreviated fitting function list and a less complex interface than the advanced mode. Additionally, the basic mode offers less control over the fit and less customization of the reported results.



The Advanced Mode

The advanced mode lets you customize all aspects of the fitting process. It provides access to many more fitting functions than the basic mode and the functions are separated into categories to facilitate searching. The advanced mode also has its own menu and toolbar to provide access to all its features.

To select a function in the advanced mode, first select Function:Select from the NLSF menu if the Select Function page is not active. Then select the appropriate category from the Categories list box, and select the desired function from the Functions list box. Once a function is selected, the procedure for fitting is the same as for fitting after you define your own function (see the following sections).



Defining a New Fitting Function

In this tutorial, we will define and compile a new fitting function using Origin C, Origin's built-in programming language. We could also define the fitting function using LabTalk script, Origin's built-in scripting language. However, defining the function using Origin C delivers more in terms of flexibility and speed (up to 20X faster than LabTalk).

To Open the Project File:

27. On the Standard toolbar, click the Open button  .
28. In the \Tutorial folder, open ASYMGAUSS.OPJ. The project opens showing a worksheet containing data and a graph window containing a data plot.

We will define a new function for fitting data with an asymmetric Gaussian peak shape. An asymmetric Gaussian is a peak function characterized by different widths on either side of the peak center, and is defined as:

$$y = y_0 + \frac{A}{\frac{(\sigma_1 + \sigma_2)}{2}\sqrt{2\pi}} \exp\left(-\frac{(x - x_c)}{2\sigma_1^2}\right)$$

for $x \leq xc$

and

$$y = y_0 + \frac{A}{\frac{(\sigma_1 + \sigma_2)}{2}\sqrt{2\pi}} \exp\left(-\frac{(x - x_c)}{2\sigma_2^2}\right)$$

for $x > xc$

The basic mode of the NLSF only permits defining new functions with LabTalk script. We will use the NLSF's advanced mode so that we can use Origin C.

To define your own Function in Advanced Mode:

29. From the menu, choose Analysis:Nonlinear Curve Fit:Advanced Fitting Tool. This opens the NLSF.
30. If the NLSF opens in basic mode, click **More** to switch to advanced mode.
31. From the NLSF menu bar, select **Category:New**. This opens the **Category Name** dialog box.
32. Type in:

    Tutorial Example

    Click **OK**.

Tutorial Example is added to the **Categories** list.

33. From the NLSF menu bar, select **Function:New**. This opens the **Define New Function** dialog box.
34. In the **Name** text box, type the following:

    AsymmetricGaussian

Note: Spaces are permissible in category names but **not** in function names.

35. Select the User Defined Param. Names check box.
36. In the **Parameter Names** text box, enter:

    ```
    y0,A, xc, sig1, sig2
    ```

37. Verify that the **Use Origin C** check box is checked, and click the **Edit in Code Builder** button. The **Code Builder** IDE opens with a specialized window intended for defining new fitting functions.
38. In the editable part of the Code Builder workspace, type in the following lines of code (You can copy the code from this tutorial and paste it into Code Builder. The tab formatting will be off, but the function should compile correctly):

```
double B = A / (0.5 * (sig1 + sig2) * sqrt(2 * pi));
if( x < xc )
                                y = y0 + B * exp(-0.5 * ((x - xc) / sig1) ^ 2);
else
                                y = y0 + B * exp(-0.5 * ((x - xc) / sig2) ^ 2);
```

39. Click the **Compile** button to compile the function code. Origin types **Done!** to the Code Builder output window.
40. Click the **Return to NLSF** button. This returns you to the NLSF tool. Note that our new fitting function registers in the **Definition** text box.
41. Click **Save** to save the function.

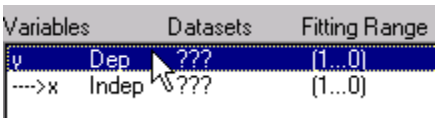
The function is saved under the name AsymmetricGaussian. This function will now be available in the list of functions under the Tutorial Example category. To verify this: From the NLSF menu, select **Function:Select**. To return to the current view, select **Function:Edit**.

Assigning the Function Variables to the Data Sets

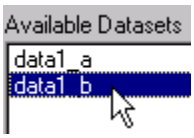
The next step is to assign the independent and dependent variables to the corresponding X and Y data sets.

To assign the function variables to the data sets:

42. From the NLSF menu, select **Action:Dataset**. This opens the **Select Dataset** dialog box.
43. Select the y variable in the top list box.



44. In the **Available Datasets** list box, select **data1_b**.



45. Click the Assign button to the right of this list box. Note that the x and y data sets are assigned to the dependent and independent variables (top list box).

How did Origin know to assign the data set **data1_a** to the independent variable x? Origin assumes that the y data set should be associated with the nearest x data set to the left of the y data set. While Origin can work with other arrangements, this is the general, default Origin behavior.

Assigning Initial Parameter Values

For the iterative fitting process to succeed, the NLSF must have reasonably good initial parameter estimates. You can manually assign initial values or you can write code that analyzes your data and performs some parameter initialization. You can also simulate the fit curve by trying various parameter values; when the simulated curve and raw data plot converge, you can assume that you have arrived at reasonable initial parameter values.

Note: Origin provides data set-specific parameter initialization for all built-in fitting functions.

In this tutorial, we will manually assign the initial values during the fitting session. For more information on simulating the fit curve, search on "simulate" in the Origin help file (**Help:Origin**). For further information on writing your own parameter initialization code, search on "parameter initialization."

Fitting the Data

We are now ready to fit our data using our AsymmetricGaussian function.

To fit the data:

46. From the NLSF menu, select **Action:Fit**. This activates the **Fitting Session** dialog box.
47. Type in the following initial parameter values:

    y0: **10**

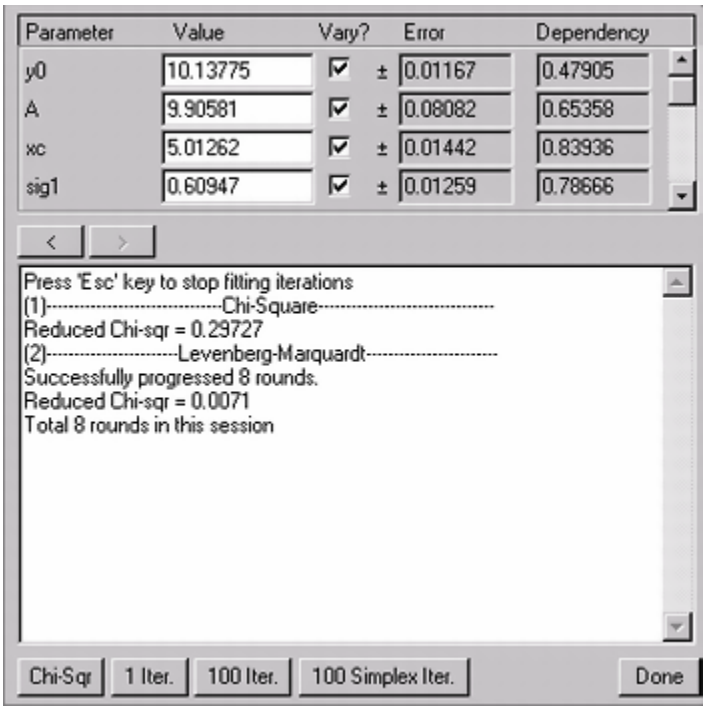
    A: **10**

```
xc:      5
sig1:    1
sig2:    1
```

48. Click the **Chi-Sqr** button. The reduced chi-squared value for the current parameter values displays in the view box.
49. Click the **100 Iter.** button.

Origin fits the data, performing a maximum of 100 Levenberg-Marquardt iterations. The fit curve displays in the graph. The reduced chi-squared value and the number of iterations performed are reported in the NLSF view box. The updated parameter values are shown in the Value text boxes.

Note: You can also define your initial parameter values in the **Parameter Initializations** dialog box (from the NLSF menu, **Scripts:Parameter Initialization**), and they will be saved with the .FDF file.



Post Fit Output Options

After fitting your data, you can opt to create a worksheet that contains the results of your fitting session. Additionally, you can have your curve fit parameters written to the results log or in a label pasted to the graph window.

To create a worksheet with the fitting results:

50. From the NLSF menu, select **Action:Results**.
51. In the **Other Options** group, click the **Param. Worksheet** button. Origin creates worksheet named **Parameters1**. The worksheet contains the curve fit results including parameter values, confidence limits, r-square value, etc.

To write parameter values to the results log and paste them to the graph window:

52. From the NLSF menu, select **Scripts:After Fit**.

53. Make sure that the Write Parameters to Results Log and Paste Parameters to Plot check boxes are selected.
54. Click the **Close** button  in the upper right hand corner of the NLSF.

The parameter values are pasted to the graph of the raw data and the best-fit curve. Parameter values are also written to the Results Log.

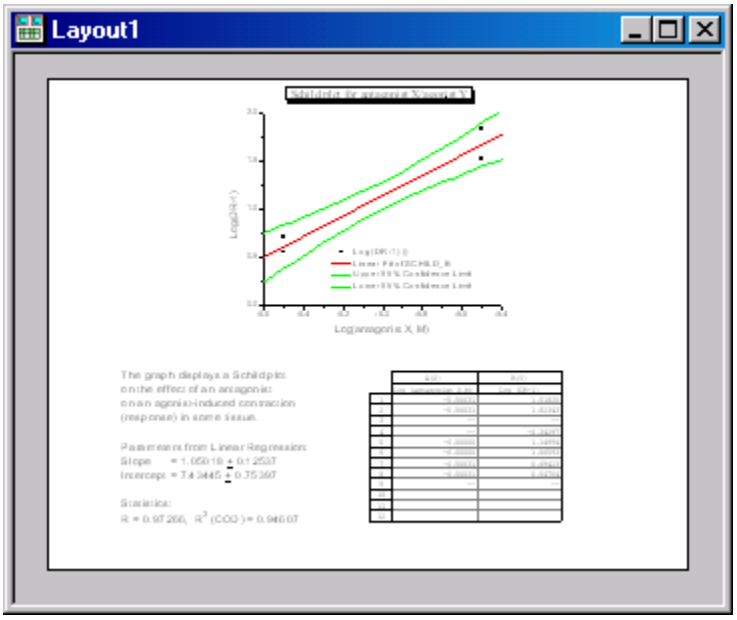
This concludes **the Nonlinear Curve Fitting** tutorial.

Tutorial 8, Creating Presentations with the Layout Page

Origin's Layout window provides a "canvas" for displaying and arranging pictures of worksheets and graphs from other windows in your project, as well as text and other annotations. The Layout window is the only means for presenting worksheet data along with your graph.

Introduction

This tutorial will show you how to add and arrange various elements on a layout page, how to edit elements of the layout page and how to export the layout page to one of Origin's many graphic export file types.



Adding Graphs, Worksheets and Text to the Layout Page

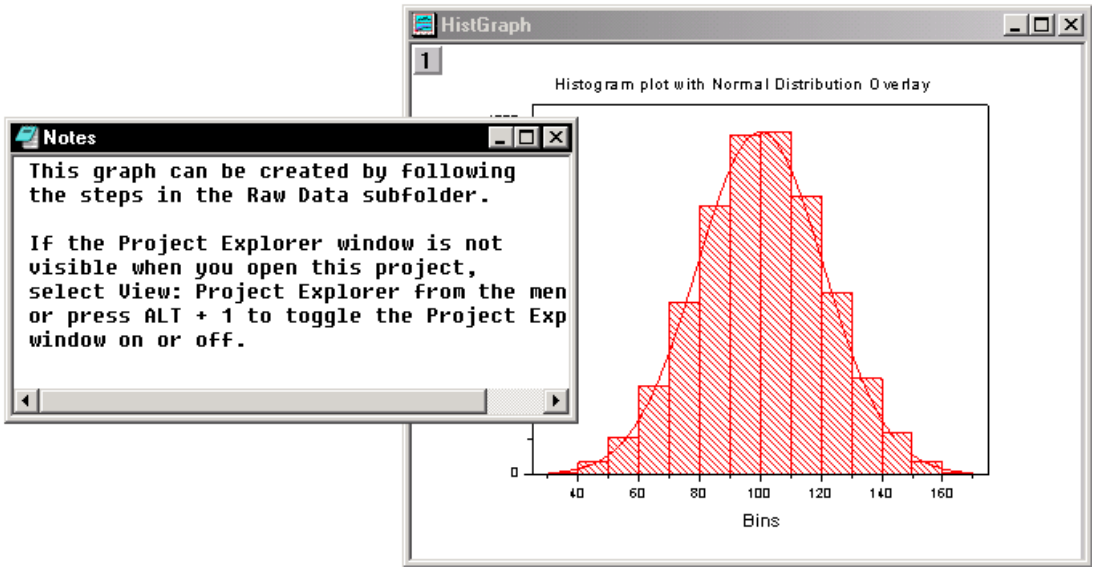
Pictures of graphs and worksheets are added to the layout page by clicking the buttons on the Layout toolbar, or by selecting associated menu commands. Text can be added with the Text tool, or by pasting from the Clipboard. Shapes, lines, and arrows can be added using the drawing tools from the Tools toolbar.

The data for this tutorial lesson is provided in an Origin project file.

To open the project file:

1. Click the **Open** button  on the Standard toolbar.
2. In the Origin \Samples\Graphing\Statistical Graphs subfolder, select Histogram.OPJ from the list of files.
3. Click **Open**.

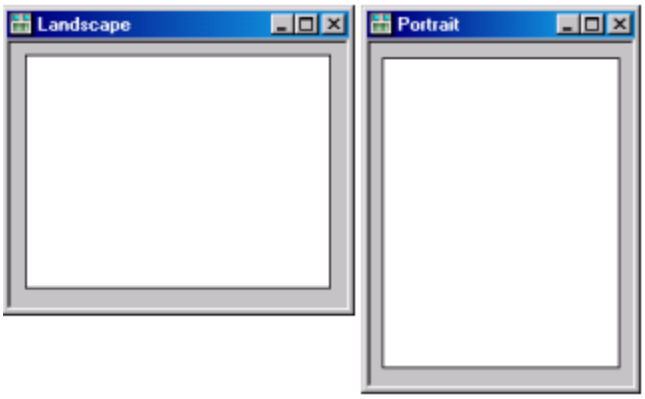
The Origin project file opens. You should see a notes window and a histogram in a graph window. The project file contains other windows in another folder. Recall that the Project Explorer workspace is toggled on and off by pressing ALT+1.



Creating a New Layout Page

To create a newLayout page:

4. Click the **New Layout** button  on the Standard toolbar. A blank layout page opens.



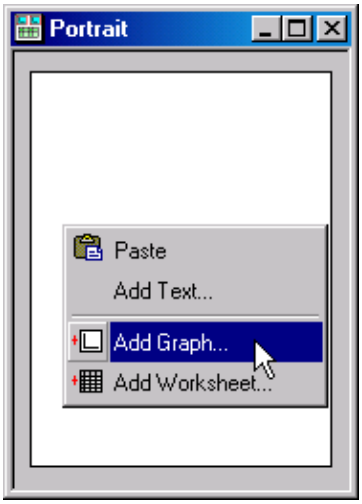
5. If the layout page is displayed in landscape orientation, right-click on the gray area just outside of the layout page (but inside the layout window), and select **Rotate Page** from the shortcut menu. The layout page will now display in portrait orientation.

Adding Pictures and Text to a Layout Page

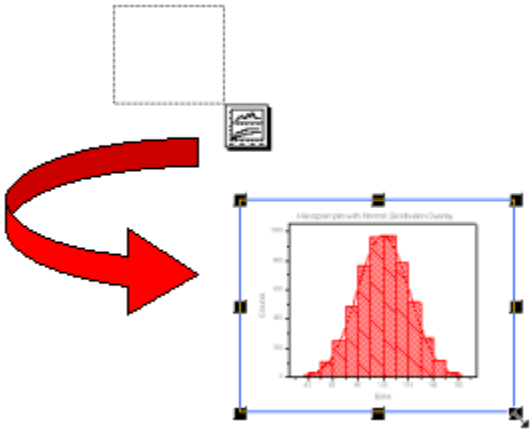
You can add pictures of any worksheet or graph from your current project to a layout page. Remember as you work with these that they are *only pictures*. Graphs and worksheets in the layout window are not directly editable. If you want to change some details of the picture subsequent to adding it to your layout window (a line color or worksheet cell value, for example), you must make the change in the source window. After making your changes, activate the layout window and click the **Refresh** button  . This will update the pictures in the layout window.

To add pictures of graphs and worksheets to the layout page:

6. Right-click in the layout page (do not click in the gray area just outside the page) and select **Add Graph** from the shortcut menu.

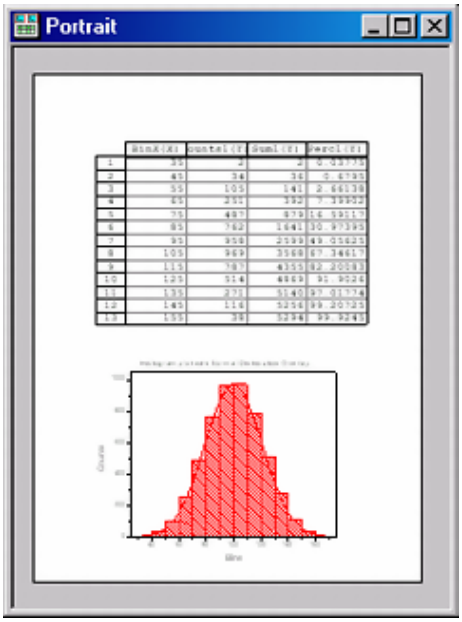


7. Select **HistGraph** from list box in the **Select Graph Object** dialog box and click **OK**.
8. Drag out a box on the bottom half of the layout page. After you release the mouse button, a picture of the HistGraph graph displays in the layout page.



When the graph picture is selected, you can drag the picture to a new location or use the sizing handles to resize it.

9. Right-click on a blank section of the layout page and select **Add Worksheet** from the shortcut menu.
10. Select **Bin1** from the list box in the **Select Worksheet Object** dialog box and click **OK**.
11. Drag out a box in the top half of the layout page, leaving some space at the top for a title.



After you add the worksheet picture to the layout page, it may display a blank area to the right of the last worksheet column, and it may include a number of blank worksheet rows.

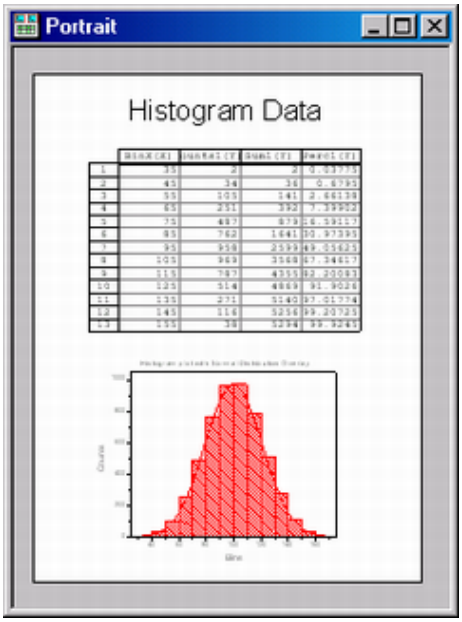
To remove blank areas from the worksheet picture:

12. Right-click on the worksheet picture in the layout page and select **Keep Aspect Ratio** from the shortcut menu. This cancels the selection.
13. Drag one of the control handles on the right edge of the worksheet picture so that only the four worksheet columns display.
14. Drag one of the control handles on the bottom edge of the worksheet picture so that only the worksheet rows with values display.
15. Re-position the worksheet picture in the layout page.
16. Click outside the worksheet picture (but within the layout page) to cancel the selection.

To add text to the layout page using the text tool:

17. On the **Tools** toolbar, click the **Text Tool** button  .
18. Click above the worksheet picture in the layout page.
19. Before you start typing, from the **Format** toolbar, select **36** from the **Font Size** combo box.

20. Type **Histogram Data** in your new text label.
21. Click outside the label to cancel text editing mode.



Customizing the Appearance of the Layout Page

In this section, you will fine tune the position of the pictures displayed in the layout page. In addition, you will make changes to the source graph window to change the picture's appearance in the layout page.

There are several ways to arrange pictures on the layout page. You can drag the pictures and "eyeball" the position, use the Object Edit toolbar to align and group objects, or you can view the layout page grid (**View:Show Grid**) and align the pictures using the grid lines as a guide.

To arrange the pictures on the Layout page using the Object Edit tools:

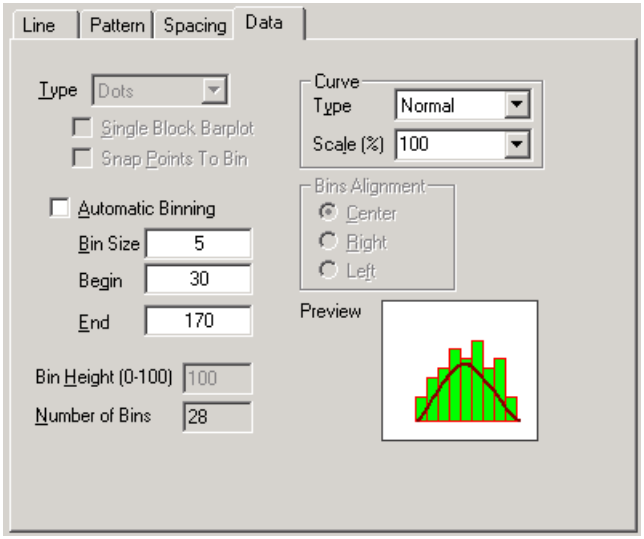
22. If the Object Edit toolbar is not currently open, select **View:Toolbars**. From the **Toolbars** tab, select the **Object Edit** check box, then click **Close**.
23. Click on the **Histogram Data** text label to select it.
24. Press SHIFT and click on the worksheet picture, and then on the graph picture (in that order). All three objects are now selected in the layout page.
25. Click the **Vertical** button  on the Object Edit Toolbar. The pictures align vertically on the last selected picture (the graph).
26. Click in the layout page but outside of the pictures to clear the selection.

Editing the Pictures in the Layout Page

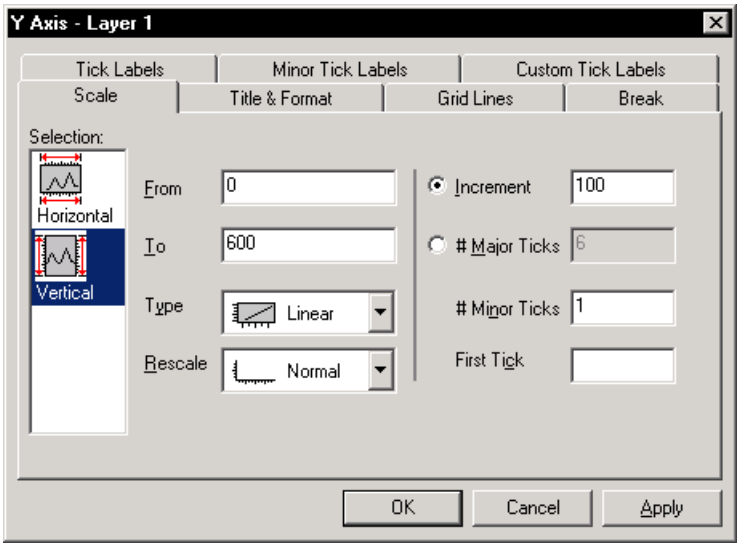
As mentioned, you do not edit the worksheet and graph pictures directly in the layout page. Instead, you edit the source window and update the pictures in the layout window. Origin provides a shortcut menu command to take you from a worksheet or graph picture to the source window.

To edit a source window from the Layout page:

27. Right-click on the histogram picture in the layout page.
28. From the shortcut menu, select **Go To Window**. Origin activates the HistGraph window.
29. Right-click on the histogram and select **Plot Details** from the shortcut menu. The **Plot Details** dialog box opens.
30. Select the **Data** tab.
31. Clear the **Automatic Binning** check box.
32. In the **Bin Size** combo box , type **5**. Click **OK**.



33. Double-click on the Y axis in the HistGraph window. This opens the **Y Axis** dialog box.
34. Select the **Scale** tab (if not already selected).
35. In the **To** text box Type **600;** in the **Increment** text box type **100**. Click **OK**.



36. Activate the layout page window.
37. Click the **Refresh** button on the Standard toolbar.

Exporting the Layout Page

Once you have tweaked your layout page presentation, you might wish to print it out for a poster presentation or you might export the layout page to another application.

You can export your graph or layout page by copying and pasting or you may opt to use any one of a number of built-in raster and vector graphic export filters to create a separate image file.

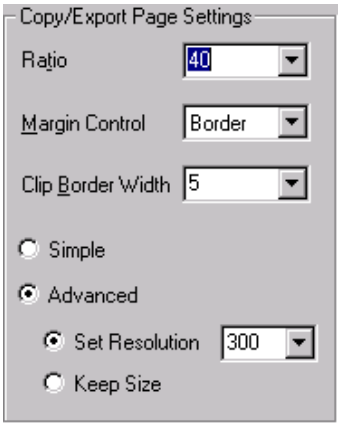
Before you export your layout window we should say something about the settings that control the size of the exported image.

- For *Clipboard* (copied) and *vector* files types, image dimensions are controlled by (1) page size (default is printer page size) and (2) the settings in the Copy/Export Page Settings group on the Page tab of the Options dialog box (from the menu, select **Tools:Options**).
- For *raster* file types, image dimensions are a function of (1) page size and (2) export file resolution (commonly between 72 and 600 dpi). In effect, raster files are dimensionless because when you multiply page size by dpi, you get only "dots;" the dimensions cancel out. Thus the dimensions of a raster file are controlled by the resolution of the rendering device, be it a 96 dpi video display or a 1200 dpi printer.

Note: Common vector file types are EPS, PDF, WMF, EMF, AI, CGM, and DXF. Vector files are often preferred by print publications because they can be resized without loss of image quality. These formats generally create moderate file sizes. Common raster formats are BMP, TIF, JPG, GIF, PNG, and TGA. Raster formats are good for reproducing graphs with considerable detail and color variation. They are not highly scaleable; rescaling often leads to loss of detail and color. These formats can produce a very large file size when high resolutions and colors are needed.

Since we are going to be exporting a vector file shortly, let's open the Options dialog box and have a look at the settings on the Page tab.

38. From the menu, select **Tools:Options.**
39. In the **Options** dialog box, select the **Page** tab.



As mentioned, the settings in the Copy/Export group control the dimensions of Clipboard and vector images. The default settings are pictured here. In most cases, you will only need to concern yourself with the Ratio, Margin Control, and Clip Border Width settings. Briefly:

- The **Ratio** combo box specifies the dimensions of your exported image as a percentage of your graph page size. You can choose a number from the associated drop-down list or you can type directly in the text box. A ratio of 100 will create an export image that has the same dimensions as your graph page; a ratio of 50 will produce and image that is half of your page size, etc.
- The **Margin Control** list allows you to specify how much of a margin (surrounding white space) to include with your graph or layout page pictures. The choices are **Border**, **Tight** and **Page**. If you

choose **Border**, Origin will use the **Clip Border Width** setting to determine margin width (see the next item for an explanation). If you choose **Tight**, Origin will draw the smallest rectangle that it can around your all objects in your graph or layout page and export only what is inside that rectangle. This usually minimizes white space. If you choose **Page**, Origin will export the entire page – that is, whatever is within the white background area of your page or layout window. This maximizes white space

- The Clip Border Width combo box, in concert with the **Margin Control = Border**, specifies how much of a margin to include with graph or layout page objects, as a percentage of page size.

Attention to Ratio and Margin Control/Border width settings will suffice for most situations. You are advised to leave the **Advanced** radio button selected and **Set Resolution** at 300 (or perhaps 600). However, you are encouraged to experiment with these settings, as you have time, so that you will better understand how they affect the exported image.

You should know that image export can be a fairly complex topic and there is much more to say about it than we can say here. For more information, select **Help:Origin** and search on **exporting your graph or layout page**.

Returning to our tutorial, we will export our histogram layout page as an EPS file. Encapsulated PostScript files are highly scalable; that is, they can be resized without loss of detail. For this reason, they are a sensible choice for print publishing.

To export the layout page as an Encapsulated PostScript file:

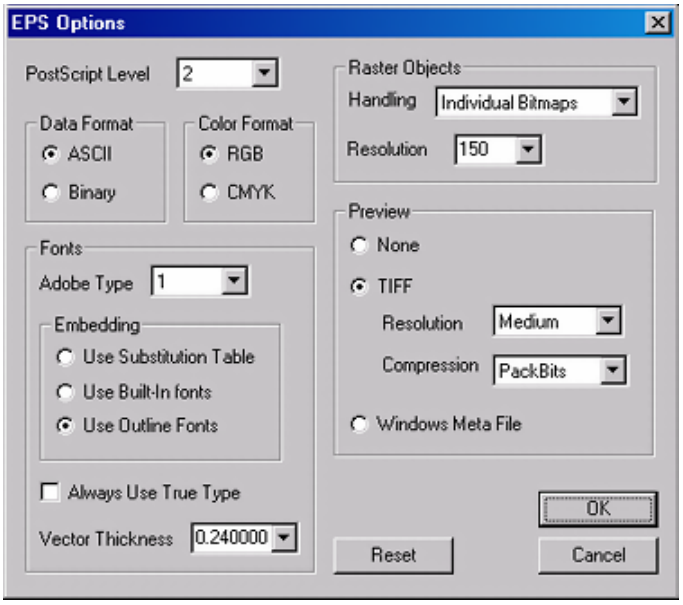
40. Make certain that your histogram layout page is the active window. From the menu, select **File:Export Page**. The **Save As** dialog box opens.
41. In the **File Name** text box, type **Presentation1**.
42. In the Save as type: list, select Encapsulated PostScript (*.EPS).

Each export file type has its own set of Export Options (pertaining to resolution, color density, etc.). If you want to view or modify those export options, you must select the **Show Export Options** dialog box at the bottom of the Save As dialog box.

43. Select the **Show Export Options** check box.
44. Specify a **Save in** folder (the Origin software folder is fine) and click **Save**.

This opens a message reminding you that the Copy/Export page settings will be used to determine the size and the margins of your EPS image.

45. Click **OK** to close the reminder message. The **EPS Options** dialog box opens.



In most cases, these default settings will work well, but you may find that changing one or more of them is sometimes necessary (in doing this, you will create new default settings).

Note: When you have determined that your export options are working satisfactorily, you may clear the **Show Export Options** check box in the **Save As** dialog box, and bypass the export options dialog. Export options are stored for *each* export file type.

One thing that can be handy when working with EPS files, is to create a TIFF (.TIF) preview. Some applications do not display EPS files and will instead display a picture placeholder. If you opt for a TIFF preview, the receiving application can use the appended TIF file to display your image.

46. In the **Preview** group, select the **TIFF** radio button if it is not currently selected.
47. Click **OK**.

The layout page is exported as an EPS file and can be inserted into any application that recognizes this file type. If you have Adobe Photoshop or Ghostview, you should open your newly created EPS file and have a look.

This concludes this tutorial on **Creating Pesentations with the Layout Page**.

Command Line Tutorials

The Command Line Tutorials will show how you can issue LabTalk commands from the Script Window to return worksheet values, perform math operations on datasets and do such things as change plot attributes. You will also learn how to define and call a LabTalk macro.

Tutorial 9, The Origin Script Window

The Script Window provides a convenient way to send LabTalk script commands to Origin. In this tutorial you will use the Script Window to do simple calculations, read and write to Origin worksheets and change plot attributes in a graph window.

Performing Calculations

One of the more basic uses for the Script Window is as an interface to perform simple calculations.

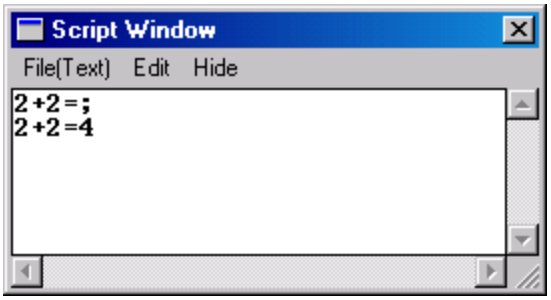
1. From the Origin program menu, choose **Window:Script Window**. The Script window opens.
2. Type the following in the Script window:

   ```
   2+2=
   ```

   Press **ENTER**.

Origin appends a semi-colon the end of the line and returns:

```
2+2=4
```



3. Type the following:

   ```
   log(100)=
   ```

   Press **ENTER**.

Origin appends a semi-colon at the end of the line and returns:

```
LOG(100)=2
```

Each statement in a LabTalk script must end in a semi-colon. However, if your intent is to execute a single line of code from the Script window, you *do not type the semi-colon at the end of the line* . Simply press **ENTER** to complete the line. Origin will append a semi-colon for you and execute the Script command.

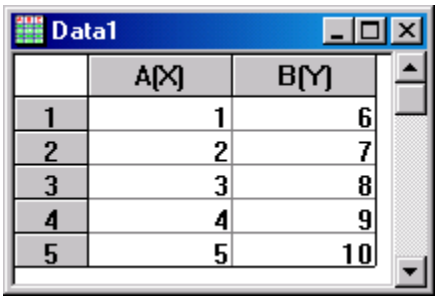
If you are typing multiple lines of code in the Script window and you do not want code to execute each time you enter a carriage return, then you should type the semi-colon at line's end and press **CTRL + ENTER**. To execute multiple lines of script from the Script window, verify that the Script window's Edit:Script Execution menu command is checked, select the entire script and press **ENTER.**

Note: Any mathematical function – built-in as well as user-created – can be executed from the Script Window.

Reading and Writing Worksheet Values

You can also use the Script Window to read and write worksheet values, or to perform math operations on datasets.

Enter the following data into a fresh worksheet:



	A(X)	B(Y)
1	1	6
2	2	7
3	3	8
4	4	9
5	5	10

4. To return the value in the first cell of the second column, type the following:

```
cell(1,2)=
```

Press **ENTER**.

Origin returns:

```
CELL(1,2)=6
```

You can also use the column name and row number to reference cell values.

5. Type the following:

```
col(B)[1]=
```

Press **ENTER**.

Origin returns:

```
COL(B)[1]=6
```

Note: In addition to using the column name, you can also use the dataset name. In LabTalk, the syntax for naming datasets is `worksheetName_columnName`. So, For example, `Data1_A[1]=` would return the first element of column `A` in worksheet `Data1.` Also, if the worksheet that you are referencing is the active window, you can use the LabTalk string variable `%H`, in place of the worksheet name. For example, `%H_A[1]`.

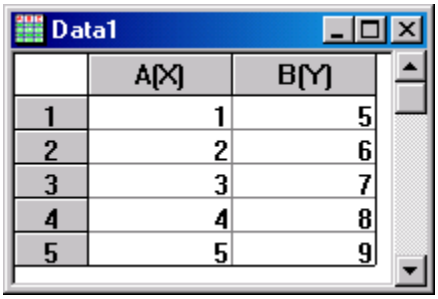
To subtract the value in row 1 of column A, from all the values in column B…

6. Type the following:

```
col(B)=col(B)-col(A)[1]
```

Press **ENTER**.

Your worksheet now reads:



Data1

	A(X)	B(Y)
1	1	5
2	2	6
3	3	7
4	4	8
5	5	9

Let's use what we have learned about executing multiple lines of script in the Script window. We will multiply every value in a column of data by some constant b.

7. Type the following:

```
b=3;
```

Press **CTRL+ENTER**.

Recall that this gives us a carriage return without executing the command.

8. Now type:

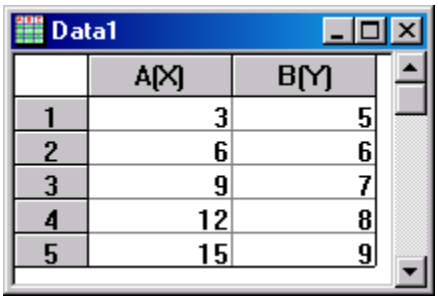
```
col(A)=col(A)*b;
```

Again, Press **CTRL+ENTER**.

9. Choose **Edit** from the Script window's menu bar. **Script Execution** should have a check mark next to it; if not, single-click on the menu item to place a check mark there.

10. Now, select the two lines of script that you just entered into the Script window and press **ENTER**.

Your worksheet now reads:



Data1

	A(X)	B(Y)
1	3	5
2	6	6
3	9	7
4	12	8
5	15	9

Note: The following C notation is also supported:

```
b=3;
col(A)*=b;
```

You can also use linear interpolation or extrapolation on a specified *X dataset* to find the corresponding interpolated or extrapolated value in a *Y dataset*. This requires using a new notation with parentheses ( ) instead of brackets [ ].

In this example, `data1_b` is a Y dataset and `(4)` is a value in an X dataset (`data1_a`) for which you want to find a corresponding, interpolated Y value.

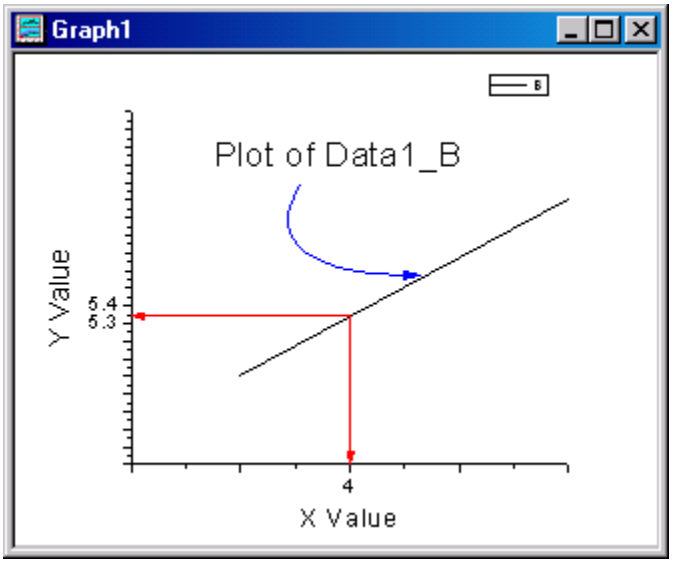
11. Type the following:

```
data1_b(4) =
```

Press **ENTER**.

Origin returns:

```
DATA1_B(4)=5.333333
```



This is a line plot of our simple worksheet data. You can see that our interpolated Y value – the one corresponding to X = 4 – is 5.333333.

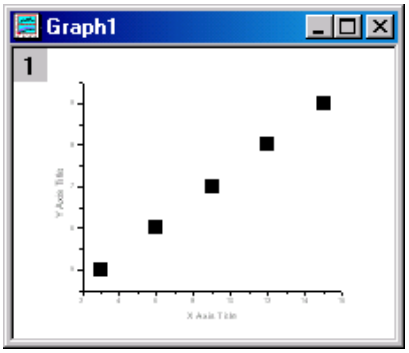
Continued in the next section, **Modifying Plot Attributes via Script**.

Modifying Plot Attributes via Script

You can use the Script Window to change the attributes of a data plot.

12. If you have not already done so, open the Script Window by selecting **Window:Script Window** from the Origin program menu.
13. Using the worksheet data from the previous exercise, create a scatter plot.

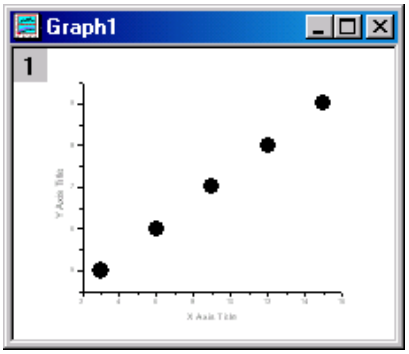
Note that the scatter plot symbol is a black filled square (symbol size is increased for clarity).



14. To change the symbol shape, type the following:

```
set %C -k 2
```

Press **ENTER**.

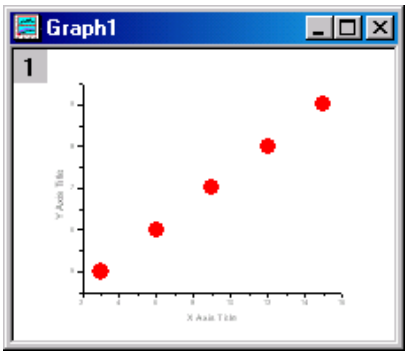


The data plot symbol changes from a filled square to a filled circle (the numbers correspond to Origin's symbol list; 1 = square, 2 = circle, etc.).

15. To change the symbol color, type the following:

```
set %C -c 2
```

Press **ENTER**



The data plot symbol color changes from black to red.

To modify the axis scale values

16. Type the following:

```
X1=0;X2=20;Y1=0;Y2=10
```

Press **ENTER**.

Your X axis scale now reads from 0-20; the Y axis reads from 0-10.

Note: As this example illustrates, you can type multiple lines of script on a single line by separating commands with a semi-colon.

You can also use the `set` *command to specify the data display range.*

17. Type the following:

```
Set %C -b 2
```

Press **ENTER**

The graph's display range now begins with the second data point in the dataset.

18. Type the following:

```
Set %C -e 4
```

Press **ENTER**

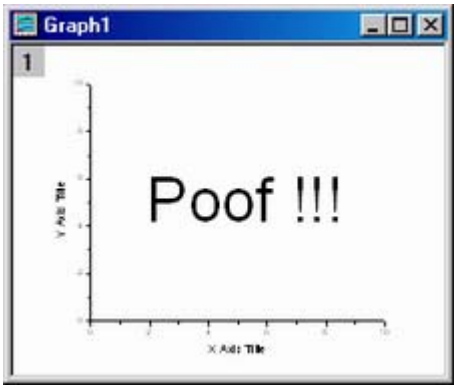
The graph's display range now ends with the fourth data point in the dataset.

You can also hide or show a data plot using the `set` *command's* `-s` *switch.*

19. Type the following:

```
Set %C -s 0
```

Press **ENTER**.



The active dataset is now hidden.

20. To show the hidden dataset, type:

```
Set %C -s 1
```

Press **ENTER**.

This concludes the **Origin Script Window** tutorial.

Tutorial 10, Adding New Commands to Origin

A macro is a convenient method of aliasing a LabTalk script. When you define a macro you are associating an entire script with a specific name. This name becomes a command that invokes the associated script.

When developing scripts, macros can provide several advantages.

- Modular code can streamline a script by replacing repetitive or similar blocks of code with multiple calls of the same macro.
- Modifications to your code become easier to implement because you only have to redefine your macro as opposed to modifying repeated blocks of code that are scattered throughout your application.
- There is a limit to the number of tokens that can be included between a set of curly braces that enclose a script. Macros provide a means to reduce the number of tokens between braces by calling pre-defined code.
- You can modify the behavior of a LabTalk command by creating a macro of the same name. The functionality of the LabTalk command is restored when the macro is deleted.

Defining a Macro

A macro is defined using the **define** command. The general syntax is,

```
define macroName
{
script
}
```

where *macroName* and *script* are the name of the macro and the body of the macro, respectively.

The define a macro using LabTalk's `define` *command:*.

1. From the Origin menu, select **Window:Script Window**.
2. Type the following:

```
def hello {
type -b "Hello World!!!";
}
```

This script defines a macro named hello that will type "Hello World!!!".

Note: The `define` command can be abbreviated as `def`

We will now use the Script window to call our `hello` macro.

3. Type the following into the Script window:

```
hello
```

Press **ENTER**.

An attention dialog opens to say "Hello World!!!".

Origin's Predefined System Macros

Let's look at Origin's predefined system macros, some of which take an argument. Macros can take up to five arguments. Use the %1, %2,… %5 notation within the script definition to indicate that the macro expects one or more arguments (%1= 1st argument, %2= 2nd argument,…%5 = 5th argument).

4. In the Script window, type the following:

```
list m
```

Press **ENTER**.

Origin responds by typing the names of predefined macros to the Script window.

0 ARRANGELAYERS

1 BEFOREITERATE

2 BEGINSAVE

3 CHECKMARGINS

4 CHECKVAR…etc.

To see the definition of any system macro:

5. To see how a system macro is defined, type **Def *macroname***. For example, type the following in the Script window:

```
Def graph
```

Press **ENTER**.

Origin responds by typing the following to the Script Window:

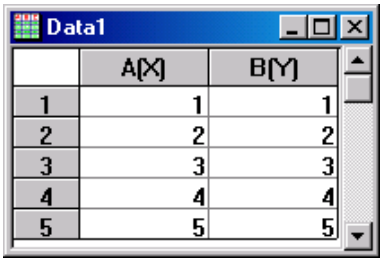
```
def graph;
{
set %1 -s 1;layer -i %1
}
```

The %1 notation in the macro definition indicates that this macro takes one argument.

Note: Recall that in Tutorial 9, we used the `set dataset -s 1` command to show a data plot. The layer -i `dataset` command adds (plots) the named data set to the active layer.

To call a macro:

6. Click the **New Worksheet** button  .
7. Create a worksheet named Data1 and type in the following data:



8. Click the **New Graph** button  .
9. In the Script window, type the following:

```
Graph data1_b
```

Press **ENTER**.

A line plot of dataset `data1_b` is included in the graph window.

We will modify the macro definition so that it creates a scatter plot using a symbol type of red, "up triangle."

You could hard code the appropriate values for scatter plot, red, and up triangle in the macro definition but it is more efficient to pass the value of a variable as an argument. In this way, the macro may be used in other instances where you want to set the color and symbol shape to something other than red, upward-pointing triangle .

Note: LabTalk often uses an integer value to specify plot details. If you look at the color palette, for instance (from the menu, **Format:Color Palette**), you will see (assuming that you have not modified the default color palette) that black =1, red = 2, green = 3, blue = 4, etc. For more information, see the `Set` command in the LabTalk Language Reference section of the Programming Help file.

10. Type the following in the Script Window:

```
def graph
{
set %1 -s 1;layer –i201 %1;
set %1 -c %2; set %1 -k %3;
}
```

11. Highlight the entire script and press **ENTER** (make certain that the Script Window menu item **Edit:Script Execution** is checked)**.**
12. Type the following:

```
graph data1_b 2 3
```

Press **ENTER**.

This plots the data set data1_b as red, upward pointing triangles.

To better understand what we actually did, let's examine our macro line by line.

First, we used the **`def`** command to tell Origin that we are defining a macro.

```
def graph
```

Secondly, we used the LabTalk **`set`** command with a **`–s`** option. The **`set –s`** command syntax is,

```
set dataset -s value
```

where *dataset* is the name of a data set, and *value* is either 1 (show plot) or 0 (hide plot),

```
{
set %1 –s 1;
```

The **layer –i#** command syntax is:

```
layer –igraphType dataset;
```

Note that the data set name has been assigned to %1.

The **set –c** command syntax is:

```
set –c color#
```

This is used to specify the plot symbol color. Note that color will be assigned to %2.

The **set –k** command syntax is:

```
set –k shape#
```

This is used to specify symbol shape. Note that symbol shape is assigned to %3.

When we execute our macro by typing,

```
graph data1_b 2 3
```

we are passing three arguments to the macro:

- **data1_b**, which is substituted for %1.
- **2**, which is substituted for %2.
- **3**, which is substituted for %3.

Note that it is merely coincidence that we chose to substitute a value of 2 for %2, and a value of 3 for %3. We could have chosen any allowed value for symbol color or shape.

Remember that any macro that you define is only available for the duration of your Origin session. If you restart Origin, you cannot execute your macro until you define it again. If you want your macro to be defined automatically when you start Origin, you can save your macro definition to Origin's MACROS.CNF file. Each time Origin starts, MACROS.CNF is read, and your macro is defined.

Note: MACROS.CNF is located in the Origin software folder. Because of a turf battle over the .CNF file extension, it will probably only be listed as MACROS and will display a terminal icon. In reality, this file is a text file and can be opened in any text editor, such as Notepad.

For more information on macros, see the LabTalk Language Reference section of the Programming Help file (**Help:Programming**).

This concludes the **Adding New Commands** tutorial.

Programming Tutorials

These tutorials are designed to give you a broad overview of the basics of Origin C, Origin's ANSI C-compliant programming language.

Tutorial 11, Introduction to Origin C and Code Builder

Origin C supports a nearly-complete ANSI C language syntax as well as a subset of C++ features including internal and DLL-extended classes. In addition, Origin C is "Origin aware." This means that Origin objects such as worksheets and graphs are mapped to classes in Origin C, allowing direct manipulation of these objects and their properties from Origin C.

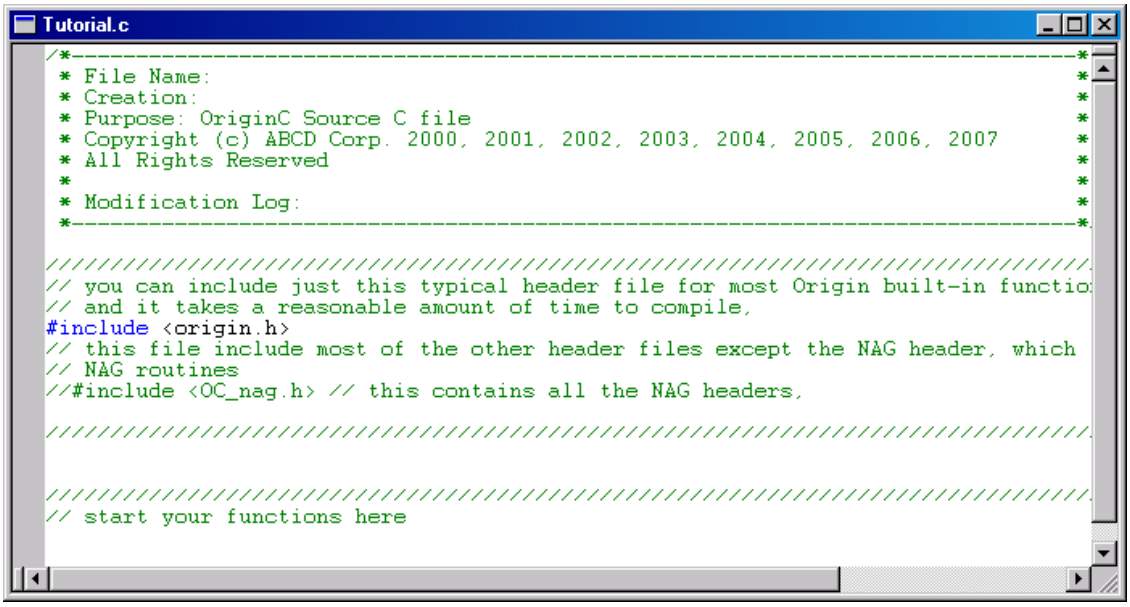
Origin C's integrated development environment (IDE) is called Code Builder. Code Builder provides standard tools for writing, compiling, and debugging your Origin C programs. Once an Origin C function is compiled, the function can be called in various ways from the Origin or Code Builder workspaces.

This tutorial introduces you to Origin C and Code Builder by showing you how to write, compile and call a function that types the message "Hello World!!!".

Hello World!!!

To begin this tutorial:

1. On the Origin Standard toolbar, click the **Code Builder** button  .
2. On the Code Builder toolbar, click the **New** button  .
3. In the **New File** dialog, select **C File**.
4. In the **File Name** text box, type **Tutorial**.
5. In the **Location** text box, select the \Origin C subfolder. Note: The Browse button looks like this:  .
6. Click **OK**. A file named Tutorial.c opens in the Code Builder workspace.



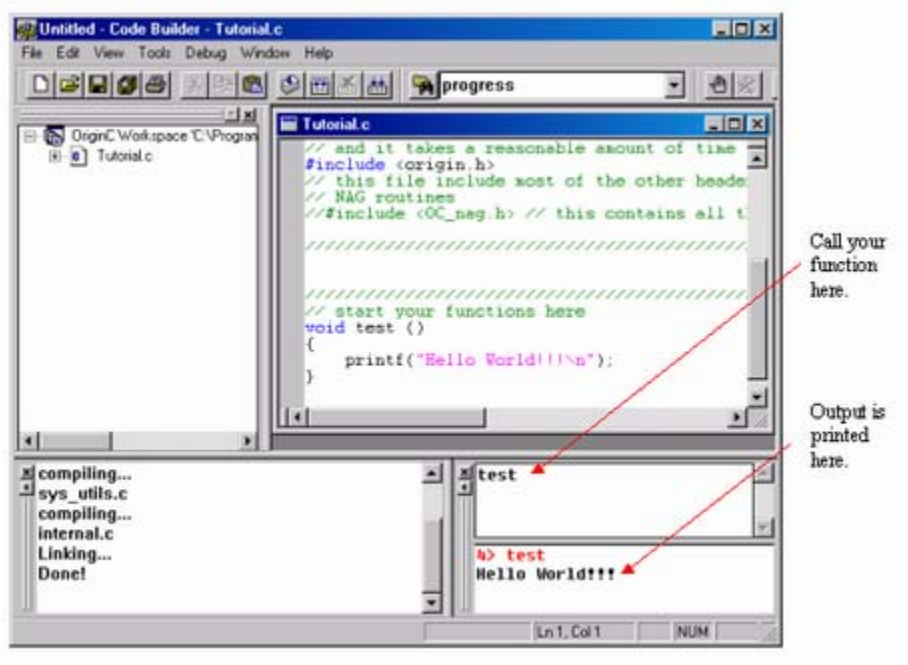
7. Type the following beneath the line that reads // start your functions here:

```
void test ()
{
                                printf("Hello World!!!\n");
}
```

8. On the Code Builder workspace Standard toolbar, click the **Build** button  . This compiles the test function.
9. To call this function, click in the upper pane of the **LabTalk Console**. This is located in the lower right corner of the **Code Builder** workspace (This is the default location. If the LabTalk console isn't visible, select **View:LabTalk Console** from the Code Builder menu and make sure that the menu item is checked).
10. Type the following LabTalk function call in the **LabTalk Console**:

    `test`

    Press **ENTER**.

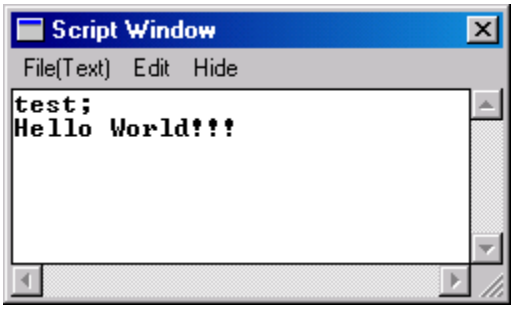


To test this function in the Origin Script Window:

11. Return to the Origin workspace, and select **Window:Script Window**.
12. In the Script window, type the following:

    ```
    test
    ```

    Press **ENTER**.



"Hello World!!!" displays in the Script Window.

This concludes the **Introduction to Origin C** tutorial.

Tutorial 12, The Code Builder Workspace

In this exercise, we will create a workspace, add a source file with a new function, then build, test and save the workspace file.

The Workspace File

A workspace is a collection of files that can be opened by a single menu option (**File:Open Workspace …**) in Origin's Code Builder. Any text file can be a part of the collection. They do not necessarily have to be source code files; they could be notes, for example.

All files opened in the Multiple Document Window by a workspace can be edited and saved individually. In addition to files being opened in the Multiple Document Window, source code files can be added to the Workspace Window with the **File:Add to Workspace** menu option.

By including source code files in the Workspace Window, you can build individual or multiple files with the appropriate menu option or tool. Header files can be referenced within source files and do not need to be loaded in the Workspace Window or even open in the Multiple Document Window.

Since you can save a workspace with a new name, you can have multiple workspace files. However, only one workspace file can be open at a time.

To create a workspace:

1. On the **Standard** toolbar, click the **Code Builder** button  .
2. From the Code Builder menu, select **File:New Workspace**. This creates a new workspace with the default name of "Untitled.ocw".
3. From the Code Builder menu, select **File:New**. This opens the **New File** dialog.
4. Choose **C File**, and type **foo** in the **File Name** text box. The **Add to Workspace** and **Fill with Default Contents** check boxes should be selected. You may accept the default Location. Click **OK**.
5. In FOO.C, starting below the line that says '\\start your functions here", type the following:

```
void bar()
{
                                printf("Hello World!\n");
}
```

6. Click the **Build** button  . Origin automatically saves the source file and compiles and links the function.
7. From the Origin menu choose **Window:Script Window**.
8. To test our new function, type:

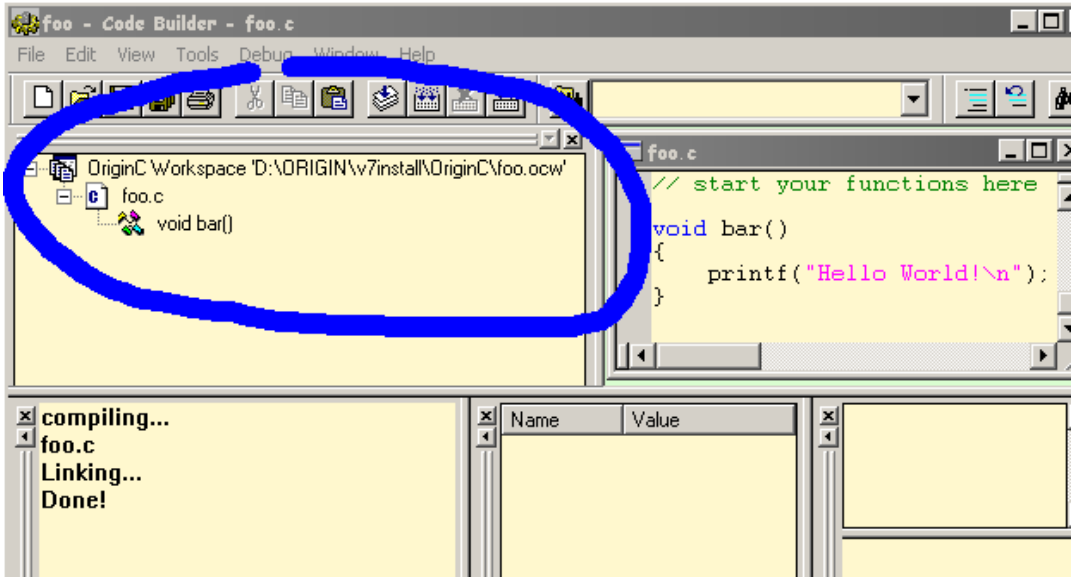
   ```
   bar
   ```

   Press **ENTER**

Origin responds by typing **Hello World!!!**

9. From the Code Builder menu, choose **File:Save Workspace As…**.

The figure shows the FOO.OCW workspace file containing a single source file, FOO.C, in the Multiple Document Window. The file has been added to the Workspace Window. The Output Window shows that the file has been compiled. The source file contains a single function – bar( ) – which is listed in the tree structure of the workspace.



System Files

Origin, itself, uses Origin C for many analysis routines. When these routines run, the Origin C source files are loaded into the Workspace Window with a special 'Temporary' status. If you right-click on the Workspace filename in the Workspace Window, you can select **Show System Files** to see what Origin has loaded (repeat the operation to hide system files). You can right-click on any System file and choose to unload it. This saves you time when you choose **Build All**.

Build on Startup

Also on the shortcut menu of the Origin C Workspace filename is the **Build on Startup** option. When this is checked, the last workspace you saved will be loaded when you restart Origin. All source files in the Workspace will be built and all functions in the source files will be available for immediate use.

For information on building individual source files on startup by including information in the ORIGIN.INI file, go to **Help:Programming** and search on ".INI."

This concludes the tutorial on the **Code Builder Workspace**.

Tutorial 13, Organizing and Accessing Origin C Functions

In this tutorial, we look at ways in which you might call your Origin C functions.

Loading and Compiling your Origin C Function

Before a function can be used, it must be compiled and linked in the current Origin session. Origin provides the following method to programmatically compile and link a source file, or to programmatically build a workspace.

```
err = run.LoadOC("myFile",[option]);
```

You can ensure that your function is compiled and linked by using the `run.LoadOC` method from your LabTalk script.

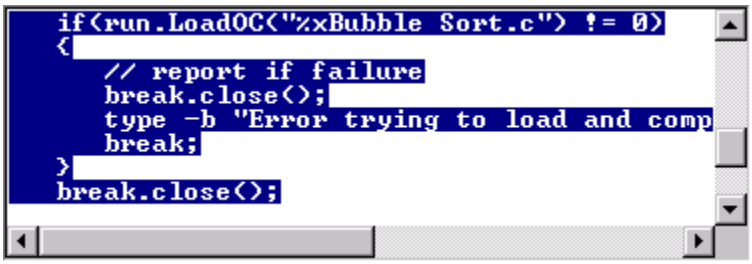
Note: For more on the LabTalk `run` object in the LabTalk Language Reference section of the Programming Help file (**Help:Programming**).

In the following example, we will create a new C file, containing a function that plots data from a worksheet. We will then program a button on a worksheet template as well as a new toolbar button, so that they both call this C function in order to create a graph.

To begin, this tutorial:

1. On the **Standard** toolbar, click the **Open** button  .
2. Browse to the \Samples\Programming\Bubble Sort subfolder in the Origin folder and open Bubble Sort.OPJ.
3. On **Graph1**(Bubble Sort Demo), ALT + double-click on the **Start** button. This opens the Label Control dialog.
4. Scroll down through the script and take note of the following line:

```
if(run.LoadOC("%xBubble Sort.c") != 0)
```



5. Click **OK**.
6. Click the **Start** button.

Notice that the Bubble Sort.c function is now 'Loading and Compiling Origin C function'. The Origin C functions in Bubble Sort.C are now accessible.

7. ALT+double-click on the **Shuffle** button to open the **Label Control** dialog box.

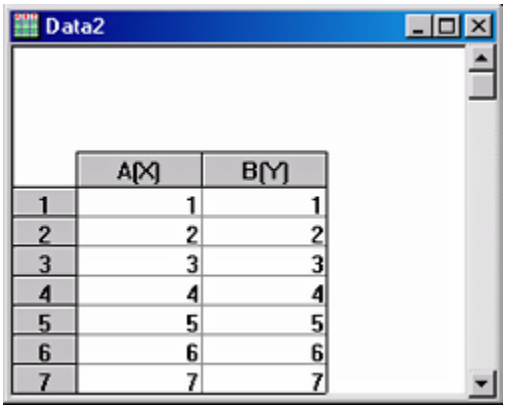
```
// The following LabTalk command just calls
// a function contained in the Origin C file
// named Bubble Sort.c
// This function shuffles the data in the
// graph so that it is arranged in random
// order. Please see the .c file for details

   bsdShuffle();
```

Because this function has been compiled, it is recognized by Origin and is ready for use.

Associating your Programs with Visual Objects

As we saw in the Bubble Sort Sample Project file, user-created visual objects – labels, arrows, lines, and other graphic elements – can be programmed to perform certain tasks. In this example, you will learn how to create a button on a worksheet and program the button to call an Origin C function.

8. Click the **New Worksheet** button  .
9. Highlight the A(X) and B(Y) columns, right-click and select **Fill Columns With:Row Numbers**.
10. From the menu, select **Format:Worksheet** to open the **Worksheet Display Control** dialog box.
11. In the Worksheet Measurement group, set the **Gap from Top** to **40**.



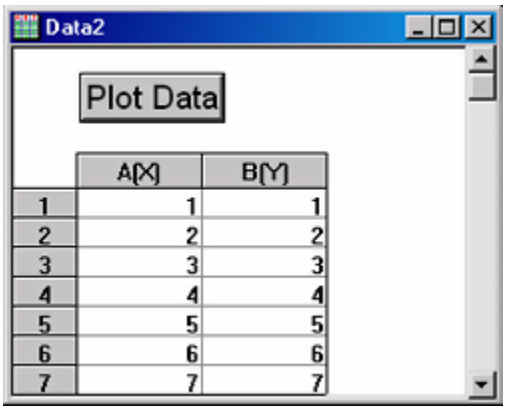
The worksheet now has sufficient space above the column headings to add a text label.

12. Right-click in the area above the two columns and choose **Add Text**.
13. At the cursor, type: **Plot Data**.
14. Click once outside the text label to deselect it.
15. Right-click on the text label and choose **Label Control**, to open the Label Control dialog box.
16. From the **Script, Run After** drop-down list, choose **Button Up**.
17. Type the following script in the text box at the bottom of the dialog box:

```
run.loadOC("test.c");
Plot_Data("scatter","data1_b");
```

18. Click **OK**.

Your text label will now look like a button.



Plotting Data into a Graph Template using Origin C

19. On the **Standard** Toolbar, click the **Code Builder** button  .
20. From the Code Builder menu, select **File:New**. This opens the **New File** dialog box.
21. In the top list-box, select **C File.**
22. In the **File Name** text box, type:

    Test.
23. Click **OK**.

The file Test.c is added to the workspace.

24. Select and copy the following function (You can copy the text in this example and paste it into CodeBuilder; the tab formatting will be off, but the function will compile correctly. Be sure to paste the text below the line that reads "//start your functions here").

```
void Plot_Data(string strTemplate, string strData)
{
                                GraphPage grph;
                                // Create a graph window from a Template
                                BOOL bOK = grph.Create(strTemplate, CREATE_VISIBLE);
                                if (!bOK)
                                    return;
                                // Attach the first layer (0) to a new GraphLayer object
                                GraphLayer grlay = grph.Layers(0);
                                Curve cv(strData);
                                //Attach a dataset to a new Curve object
                                // Add the Curve object to the GraphLayer object
                                int nPlot = grlay.AddPlot(cv);
                                if(nPlot>=0)
                                {
                                // Set plot color to Green (2)
                                grlay.DataPlots(nPlot).SetColor(2, TRUE);
                                grlay.Rescale(); // Rescale the plot
                                }
}
```

The Plot_Data function takes two arguments: (1) the template name and (2) the name of an Y dataset to include (plot) in the layer.

To call this function from the button that we created:

25. From the Origin menu, select **Edit:Button Edit Mode**. This invokes button edit mode, making it possible to click on our button object without executing any code.
26. Right-click on the button object and select **Label Control**. This opens the Label Control dialog box.
27. Type the following into the text box:

```
Plot_Data("scatter","data1_b")
```

Note that this is the Origin C function that we have just compiled.

28. Click **OK**.
29. From the Origin menu, select **Edit:Button Edit Mode**. This turns off Button Edit Mode and returns control to our button object.
30. Click the **Plot Data** button.
31. Your function creates a scatter plot with green symbols.

Note: This function assumes that you have data in column B(Y) of the Data1 worksheet and that there is an associated X data set.

In this exercise we made a call a compiled Origin C function from the Origin user interface. We could also have called this function from the Script window or from the LabTalk Console in the Code Builder IDE.

Using .OGS Files to Store Script.

As an alternative to associating your LabTalk script or Origin C function with a button, you could save your script to .OGS files. The advantage is that these .OGS files are self-contained and thus, could be called from many buttons.

These .OGS files are organized by sections. Sections are identified by a name surrounded by square brackets, as in this example:

```
[Main]
```

To execute the code in a portion of an .OGS file, you need only identify the .OGS file and refer to the containing section by name, as in this example:

```
run.section(test.ogs, Main)
```

Most of Origin's menu and toolbar commands run LabTalk script in an .OGS file. These files can be opened and edited in Code Builder.

To see the script that executes when you click an Origin toolbar button:

32. Hold down the CTRL+SHIFT keys and click the Import ASCII button  .

The file FILE.OGS opens in Code Builder. Note that it opens to the following section:

```
[ImportAscii]
```

A yellow arrow points to the first line in the [ImportAscii] section. When you click the Import ASCII button (or choose **File:Import ASCII** from the menu when a worksheet is the active window), it is this section of code – that is, the section between [Import ASCII] and [Import Lotus] – that is executed.

To create a new.OGS file:

33. From the **Code Builder** menu, select **File:New**.

34. In the **New File** dialog box, select **LabTalk Script File**.
35. In the **File Name** text box, type:

    ```
    Test
    ```

36. Click **OK**.

You now have an empty document called **test.ogs**.

You can use Origin's run.LoadOC method to programmatically compile and link an Origin C source file. The advantage of this method is that allows you to program your buttons or other user-created visual objects to make behind-the-scenes calls to your Origin C functions.

To make a call to an "uncompiled" Origin C function from Origin:

37. In the blank **test.ogs** window, type the following:

    ```
    [CreateGraph]
    run.LoadOC("test.c");
    Plot_Data("scatter","data1_b");
    ```

38. From the Code Builder menu, select **File:Save As** and save the file to your main Origin software folder.
39. Return to the Origin workspace.
40. From the Origin menu, select **View:Toolbars**. This opens the **Customize Toolbar** dialog.
41. On the **Toolbars** tab, click the **New** button to open the **New Toolbar** dialog.
42. Type in the following name for your new toolbar::

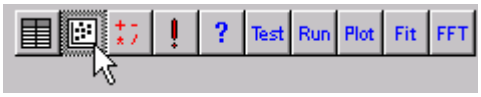
    My Toolbar

43. Click **OK**.

My Toolbar is added to the **Toolbars** list. A new toolbar is added to the Origin workspace.

Note: The toolbar may be a bit hard to see right away because it does not contain any buttons.

44. Return to the **Customize Toolbars** dialog and select the **Button Groups** tab.
45. Scroll to the bottom of the **Groups** list and select **User Defined**.
46. Select the second button in this group.

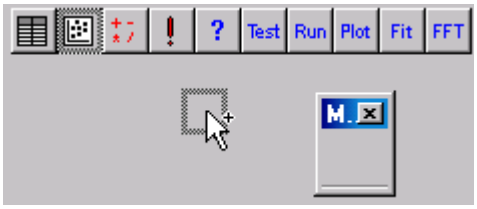


47. Click the **Settings** button. This opens the **Button Settings** dialog box
48. In the File Name text box, type **test.ogs**.
49. In the Section Name text box, type **CreateGraph**.

We are not passing arguments to the section, so we can skip the **Argument List** text box.

50. In the **Tool Tip Text** box, type **CreateGraph**. A Tool Tip is the message that displays when you mouse over a toolbar button.

51. In the **Status Bar** text box, type Example, plotting data from Origin C' for Status Bar Text. When you mouse over a toolbar button, the Status Bar message displays in the lower left corner of your Origin workspace.
52. In the **Context** group, verify that the **Windows** radio button is selected and clear the **Graph**, **Matrix**, **Layout**, and **Excel** checkboxes. Leave only **Worksheet** selected. This limits toolbar availability to active worksheets.
53. Click **OK**.
54. Point to the toolbar button, hold down the left mouse button and drag the button to the floating toolbar.



55. Click **Close** to close the **Customize Toolbar** dialog.

To test this method, close and restart Origin. Remember that the second argument to our function is data1_b, so the Origin workspace will need to have a worksheet named `data1`, a B(Y) column and some data in both the X and Y columns. Note, too, that our toolbar button is grayed out when a graph is the active window.

This concludes the tuturial on **Organizing and Accessing Origin C Functions**.

Tutorial 14, Calling NAG Functions From Origin C

Calling a NAG function from an Origin C function is very much like calling any other Origin C function. You must familiarize yourself with the desired NAG function to gain an understanding of what parameters the function requires to be passed as arguments and what parameters the function returns. Once familiar with the function, you must develop code that follows the function's requirements.

The NAG header file containing the function's prototype must be included, required parameters must be correctly declared, sized, and initialized, and the function call must follow the function's prototype as described in the Origin C Language Reference. The objective of this tutorial is to demonstrate how to call a NAG function from an Origin C function.

Understanding NAG functions

The primary resource for understanding Origin C NAG functions is the Origin C Language Reference. The Origin C Language Reference contains an entry for each NAG function that includes a description of the function, the function's prototype, the name of the NAG header file containing the function's prototype, and a description of all arguments (both input and output) required by the function. Sample data and an example program calling the function are also often included.

1. From the Origin menu select the **Help:Programming** and go to the **Origin C Language Reference** section (or from the Code Builder menu select the Help:Origin C Language Reference
2. In the **Origin C Language Reference** expand the Origin C Documentation book, expand the Global Functions book, expand the NAG Functions book, expand the NAG - Regression Analysis book, and then select the function **nag_regsn_mult_linear**.
3. Study the **nag_regsn_mult_linear** function as needed to understand the required parameters.

Secondary resources for understanding the Origin C NAG functions include the Origin C NAG header files located in the …\Origin\OriginC\system\NAG subfolder and a collection of NAG PDF files located in the \Origin\NAG PDFs subfolder.

Getting Ready to Debug Sample Code

The best way to understand how to write an Origin C function that calls a NAG function is to step through an example function in Debug mode. Follow the steps below to set up Origin and Code Builder to execute such a sample Origin C function in Debug mode.

4. On the Origin **Standard** toolbar, click the **Open** button  .
5. Browse to the \Tutorial folder and open NAGTutorial.OPJ.
6. On the **Standard** toolbar, click the **Code Builder** button  .
7. From the Code Builder menu, select **File:New Workspace**.
8. From the Code Builder **Standard** toolbar, click the **Open** toolbar button  .
9. Browse to the \Tutorial folder and open the file NAGTutorial.c.
10. In the Code Builder IDE, activate the NAGTutorial.c window (click on the window) and press **CTRL+W** to add the file to the workspace.

11. Click the Code Builder **Rebuild All** button  to compile, load, and link NAGTutorial.c.
12. From the Code Builder menu, select **View** and verify that the **LabTalk Console** (Command & Results) and the **Local Variables** windows are open (the menu item is selected).
13. From the Code Builder menu, select **Tools:Customize** and verify that the **Debug** and **Search** toolbars are selected.

Including the NAG Header

14. Activate the NAGTutorial.c window in Code Builder. Near the top of the file, above the function NAGTutorial, locate the line:

    ```
    #include <NAG\OCN_g02.h>       // NAG function
    ```

15. Right-click anywhere in the line and select **Open "NAG\OCN_g02.h"**.

This opens the header file containing the prototype for the NAG function **nag_regsn_mult_linear**.

16. In the Search combo box, type the following:

    ```
    int nag_regsn_mult_linear(
    ```

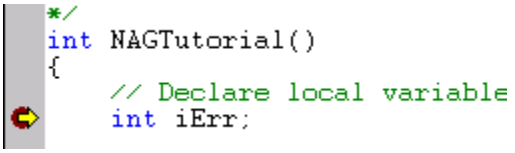
    Press **ENTER**.

The Code Builder editor scrolls to the line containing the above text. The NAG header file containing the function prototype of the NAG function you want to call must be included in your source file above the line where the NAG function is called. The name of the NAG header file containing the prototype for each NAG function is listed under the function in the Origin C Language Reference section of the Programming Help file.

Stepping Through a Sample Function in Debug Mode

17. Reactivate the NAGTutorial.c window in Code Builder. Near the top of the file, inside the function NAGTutorial.c, locate the line:

    ```
    int iErr;
    ```



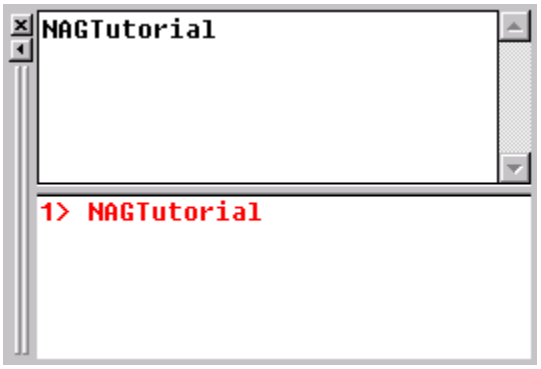
18. Position the cursor anywhere on the line.
19. Click the **Toggle Breakpoint** button  .

A brown circle displays in the gray margin to the left of the line indicating that a Debug breakpoint has been set for that line.

20. Activate the LabTalk Console (Command & Results window) in Code Builder and enter

    ```
    NAGTutorial
    ```

21. Press **ENTER** to execute the sample function having that name.

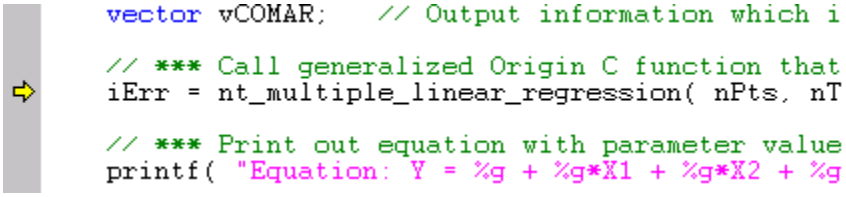


The function NAGTutorial is an Origin C wrapper function that is used to set up variables and to call a more general Origin C function named **nt_multiple_linear_regression** that in turn calls the NAG function.

22. Step through each statement in the NAGTutorial function by repeatedly pressing the Step Into button  on the Debug toolbar.

Be sure to stop along the way to read the comments for each statement. Also, periodically re-size and re-position the Local Variables window and view the current run time value of each variable as it gets set.

23. Stop when you reach the following line:

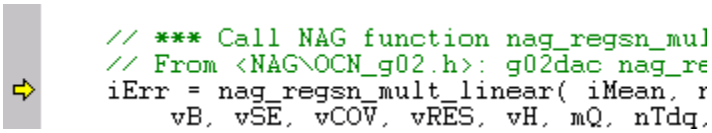


The line above calls an Origin C function named **nt_multiple_linear_regression** that in turn calls the NAG function.

24. Press the **Step Into** button  once more to step into the function **nt_multiple_linear_regression**.
25. Continue stepping through each statement in the **nt_multiple_linear_regression** function one line at a time by repeatedly pressing the **Step Into** button  on the Debug toolbar.

Be sure to stop along the way to read the comments for each statement as needed to understand the code. Also, periodically re-size and re-position the Local Variables window and view the current run time value of each variable as it gets set.

26. Stop when you reach the line that begins with the following:.



This is the line of code that calls the NAG multiple linear regression function named **nag_regsn_mult_linear**.

27. Press the **Go** button on the Debug toolbar to finish executing all code in the NAG Tutorial example.

The example functions complete execution and the equation generated by the multiple linear regression is typed out in the bottom pane of the **LabTalk Console** (Command & Results). Note that it is the same equation used to generate the sample data.

```
1> NAGTutorial
Equation: Y = -1.12347e-015 + 3*X1 + -2*X2 + 5*X3
```

This concludes the **NAG Functions** tutorial.

Tutorial 15, Accessing Internal Origin Objects

Internal Origin objects (such as Project Explorer folders, Origin windows (pages), layers, plots, graphic objects, data sets etc.) are accessed using Origin C classes. To access or programmatically control an internal Origin object you must attach it to an Origin C object.

To attach to an internal Origin object you must first "find" it using the properties, methods, and collections of a container class. Common container classes include the Project, Folder, Page, GraphPage, Layer, GraphLayer, Worksheet, MatrixLayer, and Collection classes. Once found, an internal Origin object can easily be attached to an Origin C object of the appropriate type.

The internal Origin object is then programmatically controlled by manipulating the class methods and properties of the attached Origin C object. The objective of this tutorial is to demonstrate how to find particular internal Origin objects, attach to the those objects, and to access the objects by manipulating the methods and properties of the attached Origin C objects.

Accessing Worksheet Related Objects

Familiarity with the Origin C Project class (PROJECT.H), the Collection class (COLLECTION.H), and the Folder class (FOLDER.H), is valuable when attempting to understand how to find particular internal Origin Objects. User's may find it helpful to preview these classes in the Classes book of the Origin C Help Reference or in the above header files located in the ..\Origin\OriginC\system subfolder. If you are not familiar with debugging Origin C files in Code Builder you may also find it helpful to review the Debug Tutorial before proceeding.

To begin this tutorial:

1. On the Standard toolbar, click the **New Project** button  .
2. On the Standard toolbar, click the **Code Builder** button  .
3. On the Code Builder menu, select **File:New Workspace**.
4. On the Code Builder menu, click the **Open** button  .
5. Browse to the \Tutorial folder in the Origin software directory, select AccessWorksheetObjectsTutorial.c, select the **Add to Workspace** check box and click **Open**.
6. On the Code Builder toolbar, click the **Rebuild All** button  . This compiles and links the file.
7. On the Code Builder **View** menu, verify that the **LabTalk Console** (Command & Results) and the **Local Variables** windows are visible (the menu item is checked).
8. From the Code Builder menu, select **Tools:Customize.** Select the Toolbars tab and make sure that the **Debug** toolbar check box is selected.
9. In Code Builder, activate the AccessWorksheetObjectsTutorial.c. Near the top of the file locate and click on the line:

   ```
   PageBase pb;
   ```

   You can position the cursor anywhere on the line.

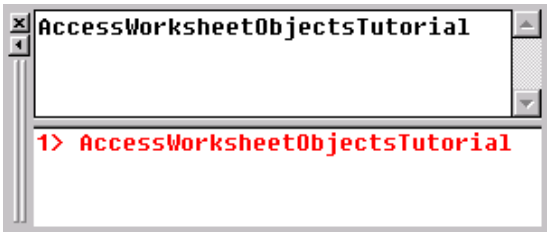
10. From the Code Builder menu, select **Debug:Toggle Breakpoints**. Alternately, press **F9** or click the **Toggle Breakpoint** button  on the **Debug** toolbar.

    A brown circle is displayed in the gray margin to the left of the line indicating that a Debug breakpoint has been set for that line.

11. In the Code Builder workspace, activate the LabTalk Console (Command & Results window) and type the following:

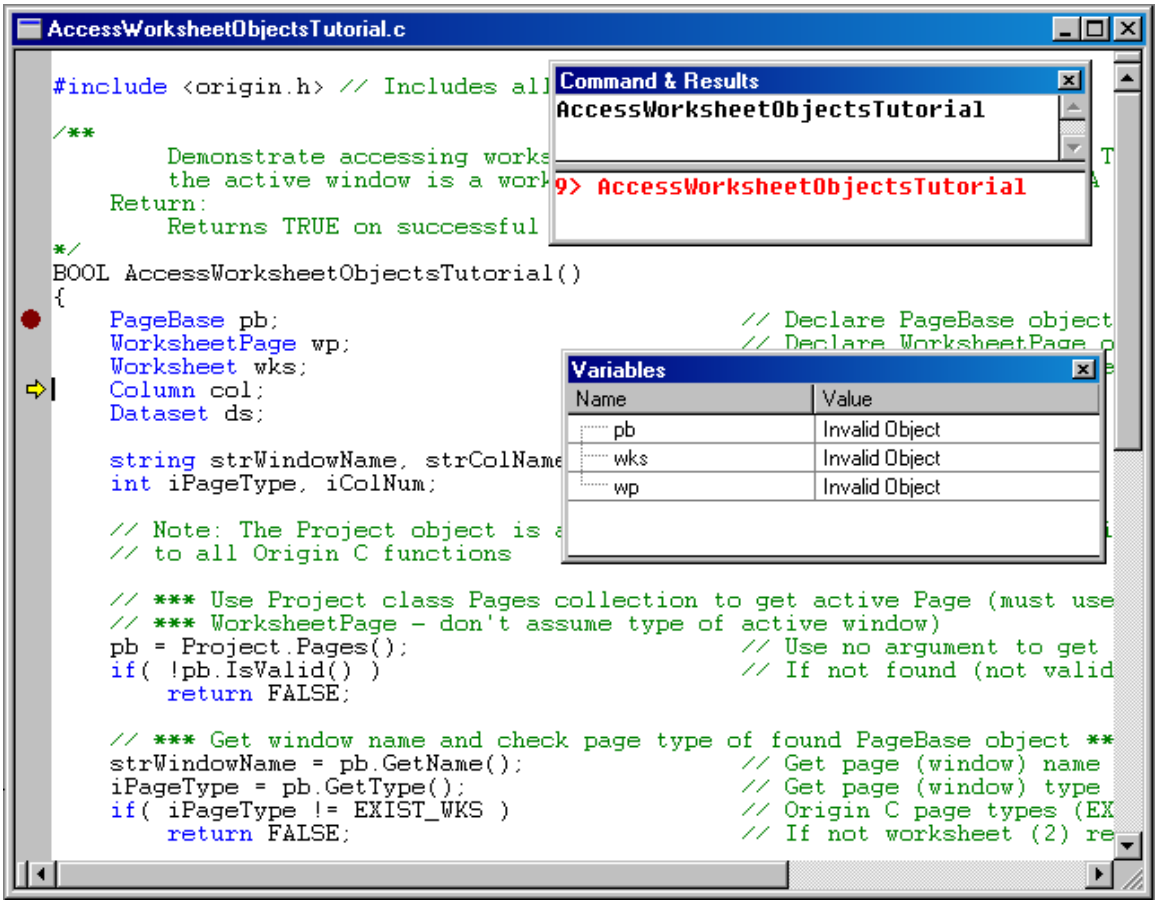
    ```
    AccessWorksheetObjectsTutorial
    ```

    Press **ENTER** to execute the function.



12. On the **Debug** toolbar, press the **Step Into** button  .

13. Press the **Step Into** button  repeatedly, stopping to read the comments for each statement. Periodically stop and re-size and/or re-position the Local Variables window to view the current runtime value of each variable.



Accessing Graph Related Objects

14. Return to the *Origin* workspace and, on the **Standard** toolbar, click the **Open** button  .
15. Browse to the \Tutorial subfolder, select AccessGraphObjectsTutorial.OPJ, and click **Open**.. You may be prompted to save changes to an untitled project. Click **No** and a worksheet and graph should open.
16. From the Code Builder menu, select **File:New Workspace.** Click **No** when prompted to save workspace changes.
17. In Code Builder, click the **Open** button  .
18. Browse to the \Tutorials subfolder, select AccessGraphObjectsTutorial.c, select the **Add to Workspace** check box, and click **Open**.
19. Click the **Rebuild All** button  to compile and link the file.
20. On the Code Builder **View** menu, verify that the **LabTalk Console** (Command & Results) and the **Local Variables** windows are visible (the menu item is checked).
21. From the Code Builder menu, select **Tools:Customize.** Select the Toolbars tab and make sure that the **Debug** toolbar check box is selected.
22. In the Code Builder workspace, activate the AccessGraphObjectsTutorial.c.

    Near the top of the file locate and click on the line:

```
GraphPage gp;
```

You can position the cursor anywhere on the line.

23. From the Code Builder menu, select **Debug:Toggle Breakpoints**. Alternately, press **F9** or click the **Toggle Breakpoint** button  on the **Debug** toolbar.

    A brown circle is displayed in the gray margin to the left of the above line indicating that a Debug breakpoint has been set for that line.

24. Activate the LabTalk Console (Command & Results window) in Code Builder and type the following:

    ```
    AccessGraphObjectsTutorial
    ```

    Press **ENTER** to execute the function.

25. On the **Debug** toolbar, press the **Step Into** button  .

26. Press the **Step Into** button  repeatedly, stopping to read the comments for each statement. Periodically stop and re-size and/or re-position the Local Variables window to view the current runtime value of each variable.

This concludes the **Accessing Internal Origin Objects** tutorial.

Tutorial 16, Debugging Origin C Files using Code Builder

Errors in source code may be thought of as falling into three broad categories: syntactic, logical, and run-time.

When you compile or "build" an Origin C source file, the compiler checks the file for syntactic errors. If the source file successfully compiles (i.e. it compiles without generating error messages), the functions in the file are rendered ready to be executed. Thus, removal of syntactic errors, while sometimes tedious, is a fairly straightforward process.

Locating logical and run time errors in your source code can be more problematic. Debugging is the process of locating logical and run time errors in a function by stepping through the source code one line (or several lines) at a time, while simultaneously examining the path of program execution and the values of variables used in the function.

This tutorial will introduce you to Origin's debugging tools and cover basic Origin C debugging techniques.

Configuring Code Builder for Debugging

Origin includes an Integrated Development Environment (IDE) named Code Builder. Code Builder is launched from Origin by clicking the **Code Builder** button  on the Standard toolbar. Code Builder is an multiple document interface (MDI) capable of opening any number of text files at one time.

In addition to the Code Builder multiple document interface (MDI), where opened files can be seen and edited, there are a number of dockable tools which perform different functions. These tools include the Workspace window, the LabTalk Console (or Command & Results window), the Output window, the Watch window, The Local Variables window, and the Call Stack. These tools can be launched from the Code Builder View menu. The objective of this tutorial is to demonstrate how several of these windows can be used together to aid in the development and debugging of Origin C programs.

How each user sets up Code Builder for debugging is, to a large extent, a matter of personal preference but there are a number of key elements. To get ready for a debug session follow the steps below.

1. Click the **Code Builder** button  on the Standard toolbar.
2. From the Code Builder menu, select **File:New Workspace**.
3. Click the **Open** toolbar button  .
4. Browse to and open the file \Tutorial\DebugTutorial.c., in the Origin program folder.
5. Activate the DebugTutorial.c window in the Code Builder workspace and press **CTRL+W** to add the file to the current Code Builder workspace.

   See Open/Added files for more information.
6. Click the **Rebuild All** button  to compile, load, and link DebugTutorial.c.
7. Select the **View** menu as needed to open the **Workspace** window, the **LabTalk Console**, the **Local Variables** window, and the **Call Stack**.

A check mark next to a View menu item indicates the window (dockable tool) is open. Docked tools can be undocked and floated by double-clicking on the two vertical bars at the left side of the window. Undocked or floating tools can be docked by double-clicking on the title bar of the tool.

8. Make sure the Debug toolbar is visible by selecting the**Tools:Customize** menu item and checking the Debug checkbox on the Toolbars tab.

The Debug toolbar can also be docked and undocked by double clicking on it.

Getting Started

9. Activate the DebugTutorial.c window in Code Builder. Near the top of the file locate and click on the line:

```
double dDelta, dTol;
```

You can position the cursor anywhere on the line.

10. Select the **Debug:Toggle Breakpoints** menu item. A brown circle is displayed in the gray margin to the left of the above line indicating that a Debug breakpoint has been set for that line.

Note: You can also toggle breakpoints on by pressing the F9 function key or clicking the Toggle Breakpoint button  on the Debug toolbar.

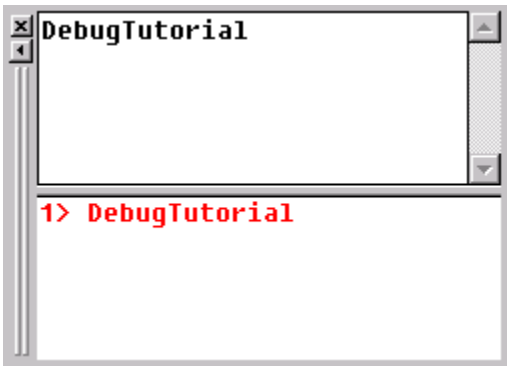
11. Toggling the breakpoint on and off a few times by repeating the last step. Leave the breakpoint on (set) before continuing.

Note: Most Debug commands can be executed by selecting a Debug menu item, by pressing special keyboard shortcut keys, or by clicking a button on the Debug toolbar. This tutorial will use each of the methods at different times but will most often use toolbar buttons. The Debug menu contains all menu items and lists equivalent shortcut keys and Debug toolbar buttons indicate functionality by icon and by tool tips activated by holding (but not clicking) the mouse cursor over the button.

12. Locate the LabTalk Console (Command & Results window) and type the name of the function that you want to debug:

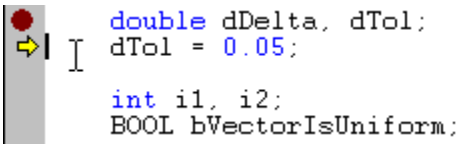
```
DebugTutorial
```

13. Press **ENTER**.



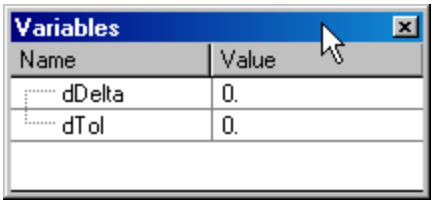
The Origin C function is executed but program execution is temporarily suspended at the line of code containing the breakpoint. Debug mode and the current point of program execution are indicated by the Debug Cursor, represented by a yellow arrow.

14. Select the **Debug:Step Into** menu item. Note that one line of code (the one containing the break point) was executed and then execution stopped again on the next line. Notice the yellow arrow has progressed to the next line indicating that line will be executed next.



15. Activate and examine the contents of the Local Variables window. Size and position the Name and Value columns by resizing the tool and by dragging the column header separation line so that the contents of both columns are completely visible.

Note that the variables dDelta and dTol are listed and their values are initialized to 0. The Path of program execution has so far only executed their declaration statement, so only these two variables are displayed.



This tutorial is continued in the next section, **Go, Step Into, Step Over, and Step Out.**

Go, Step Into, Step Over, and Step Out

16. While still in the same Debug session, click on the line:

```
i2 = vUniform.GetSize() - 1; //…
```

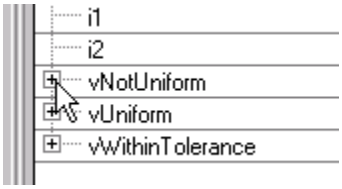
17. Press the **F9** function key to set a second breakpoint further down in the file.

18. Click the **Go** (to Origin) button  on the Debug toolbar.

The Go button causes program execution to continue to the next breakpoint in the path of execution. If there are no breakpoints in the path of execution, then the program runs until the function completes, Debug mode is exited, and control is returned to the Origin or Code Builder user interface. Since we added a breakpoint program execution stops at the next breakpoint as indicated by the new position of the Debug Cursor.

19. Activate the Local Variables window again and note the additional variables that have been added and the values that have been updated according to the path of execution.

20. Click the [+] plus sign next to the vector vNotUniform and note the vector expands showing all 21 elements and their values.



21. Activate the Watch window (**View:Watch**). Size and position the **Name** and **Value** columns by resizing the tool and by dragging the column separator so that both columns are clearly visible.

22. Double-click on the first row in the Name column and enter:

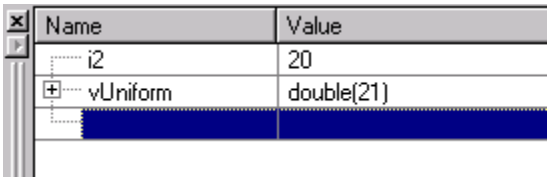
```
i2
```

23. Double-click on the second row in the Name column and enter:

```
vUniform
```

24. Click the **Step In** button  on the Debug toolbar.

Note that the values of i2 and vUniform are displayed in the Watch window. The Watch window is similar to the Local Variables window except that only entered variables in program scope are displayed.



25. With the cursor on the line:

```
bVectorIsUniform = …
```

Click the **Step In** button  once again.

Note that program control has stepped "In" to a newly called function named `VectorHasUniformSpacing`.

26. Repeatedly click the **Step In** button  , noting the path of program execution and the contents of the Local Variables window stopping after the for loop is entered but before it is exited.

Especially note blocks of code in the path of execution that get executed, skipped, or repeated depending on the result of the flow control statements (if, else, for, return, etc.) in the source code.

27. Select the **View:Call Stack** menu item and note its display of function calls and line numbers.

28. Click the **Step Out** button  on the Debug toolbar and note that program execution is passed back to the DebugTutorial function.

Note the **Call Stack** window updates.

29. Click the **Step Over** button  on the Debug toolbar three times noting that this time the function call to VectorHasUniformSpacing is not stepped into. It is executed but it is stepped "Over."

30. Click the **Go** (to Origin) button  or the **Stop Debugging** button  on the Debug toolbar. If you click the Go to Origin button the DebugTutorial function terminates normally and if you click the Stop Debugging button program execution is halted immediately.

This completes the tutorial, **Debugging Origin C Files Using Code Builder**.

Tutorial 17, Automation Example

In this simple example, we import an ASCII file, create a graph with data from the ASCII file, perform a nonlinear curve fit on the data, and export the graph with the resulting best-fit curve and a label reporting fitting results to a .BMP file.

Automating a Repetitive Routine

If your analytical work involves a lot of repetitive tasks, you might consider automating your routines. To find out how such automation routines can be linked to a toolbar button, search on "New Toolbar Buttons" in the Origin Programming Help file (**Help:Programming**). For a more elaborate automation and batch processing example, see the sample project \Samples\Programming\Automation\Automation.OPJ.

To begin this tutorial:

1. On the **Standard** toolbar, click the **Code Builder** button  .
2. From the Code Builder menu, select **File:Open**.
3. Browse to the \Tutorial folder and open **Automation.c**
4. From the Code Builder menu, select **File:Add to Workspace**.
5. On the Code Builder **Standard** toolbar, click the **Build** button  .

The Output Window reports **"Done!"** when the compile competes.

6. Return to the Origin workspace. From the menu, choose **File:New Project**.
7. From the menu choose **Window:Script Window** and type the following:

   Process %yTutorial\Sample.txt

Origin stores the software directory path in the system variable %y. The string %yTutorial\Sample.txt points to the file Sample.txt in the \Tutorial subfolder. The function Process( ) which is coded in the Automation.c reads this file name and does the following:

- The file is opened.
- A new worksheet is created.
- The ASCII file is read and the xy data are imported into the first two columns of the worksheet.
- A graph is created using this xy data.
- A nonlinear curve fit is performed on the plotted data.
- The resulting graph, with the best-fit curve and fit results, are exported to a BMP file.
- The file, named AutomationTutorial.BMP, is saved to the \Tutorial subfolder.

```
////////////////////////////////////////////////////////////////////////////////
//
// This function accepts a file name, and then does the following processing:
// 1> reads in the data and plots a new graph
// 2> performsm nonlinear fit on plotted data
// 3> exports graph with fit results to a BMP file
//
void Process(string strFileName)
{
    // Open file for processing
    stdioFile ffDataFile;
    bool bRet = ffDataFile.Open(strFileName, file::modeRead);
    if(!bRet)
    {
        printf("File not found!\n");
        return;
    }
    printf("Processing file: %s...",strFileName);

    // Create a worksheet
    Worksheet wksData;
    bool bRetW = wksData.Create();
```

This concludes the **Automation Example** tutorial.

Tutorial 18, Programming Cues to Enhance User-Friendliness

Whenever a program user takes some action, it is advisable to provide some kind of feedback to the user. If you create code that takes a while to execute, you should provide some indication that the code is working and – if possible – display partial results.

This tutorial demonstrates some standard techniques – varying the Mouse Pointer, using output messages, showing a progress box, and gradual graph updates – for providing user feedback during program execution.

The Wait Cursor

One standard Windows technique is to change the mouse cursor to a Busy cursor. Typically, this is an hourglass, but the cursor can be customized on most systems. The Busy cursor will appear over the application that created it until the application changes it back to the Normal Select cursor.

You can invoke this cursor in Origin C by using a waitCursor object.

To begin this tutorial:

1. Click the **Code Builder** button  on the Standard toolbar.
2. From the Code Builder menu, select **File:Open**, browse to the Origin\Tutorial folder and open User Friendly Interface.c.
3. To add this file to the workspace, select **File:Add to Workspace**.
4. Click the **Build** button  . The Output window reports "Done!" when the compile completes.
5. Return to the Origin workspace and open a New Project (**File:New…Project**).
6. Open the Script window (**Window:Script Window**) and type the following:

   ```
   BusyAsABee()
   ```

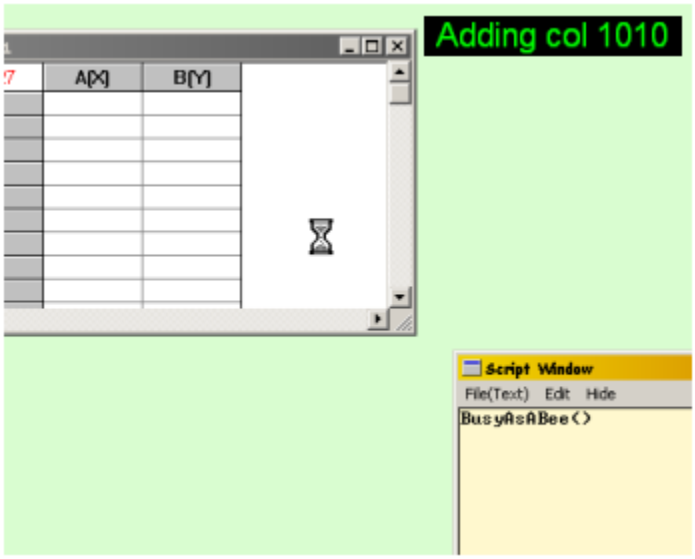
   Press **ENTER**.

When you execute the `BusyAsABee` function, you are calling the following section from the C file. Note the comments (the green text with leading //).

User Friendly Interface.c

```
// Set the Mouse Pointer to 'Busy' while long process continues
void BusyAsABee()
{
    // Create a Worksheet object from the active window
    Worksheet wks = Project.ActiveLayer();
    // If the active window is not a worksheet, create one from ORIGIN.OTW
    if(wks==NULL)
    {
        wks.Create("origin", CREATE_VISIBLE);
    }

    waitCursor  cur;        // Create a waitCursor object
    string      str;        // Create a string for messages
    int         nCol = 0;   // Create a variable to hold the number of columns added

    // Loop : 10000 times, but check for user pressing Escape key (v7SR1 or later)
    // for(int ii = 0; ii < 10000;ii++) // v7Release
    for(int ii = 0; ii < 10000 && !cur.CheckEsc();ii++)
    {
        // Add a column to the worksheet
        wks.AddCol();
        // Type a message to the Data Display window and increment column count
        str.Format("Adding col %d", nCol++);
        SetDataDisplayText(str);
    }
    // Check if the loop completed normally, otherwise the user must have pressed Escape key
    if(ii < 10000)
        printf("User aborted adding columns. %d columns were added.\n", nCol);
}
```

Origin will add columns to the active worksheet until 10000 columns have been added or you press the Escape key. While the program is executing, the Mouse Pointer will display the Busy cursor over Origin, and a message is displayed in the Data Display window indicating the current new column.



This tutorial continues with the following section, **The Progress Box**.

The Progress Box

Another programming technique that can be particularly useful when you know how many steps you will take – but you don't know how much time each step will take to execute – is to provide a progress box to indicate a percent of completion. You can use Origin C to display a Progress Box while your program executes, and update the progress indicator as needed.

This portion of the tutorial assumes that you have the "User Friendly Interface.c" file loaded and compiled from the previous example.

7. type the following in the Script Window:

```
Progress(100,0)
```

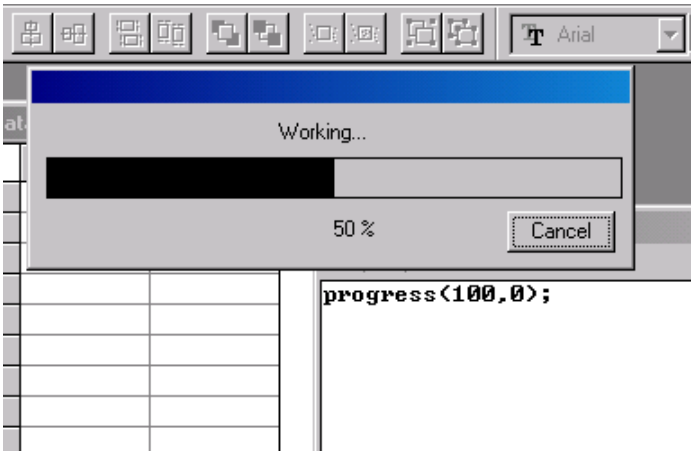
Press **ENTER**.

When you execute the `Progress` function, you are calling the following section from the C file. Note the comments (the green text with leading //).

User Friendly Interface.c

```
// Display a Progress Box
void Progress(int iBegin, int iEnd)
{
    int         iStep = 0;
    BOOL        bOK = 0;
    DWORD       dw1 = 0;
    DWORD       dw2 = 0;
    int         iDone = 0;
    int         iTemp;

    // Display the progress box
    progressBox pbox("Working...", 0);

    // Swap the arguments if the first is larger
    if(iBegin > iEnd)
    {
        iTemp = iBegin;
        iBegin = iEnd;
        iEnd = iTemp;
    }

    // Set the range
    pbox.SetRange(iBegin, iEnd);

    // Get the system tick count (in milliseconds)
    dw2 = GetTickCount();

    // Loop from the first number to the last
    for(iStep = iBegin; iStep <= iEnd; iStep++)
    {
        // Wait 1 second - This function is defined in this file
        WaitASec(1.0);
        // Set the progress indicator to the next value
        bOK = pbox.Set(iStep);
        // If the user pressed escape or clicked Cancel, then terminate the loop
        if(!bOK)
        {
            printf("Progress Box terminated by user.\n");
            return;
        }
    }
}
```

A progress box will display with a bar and number indicating percent of task completion. The percent is calculated automatically from the beginning, current and end values. Pressing the ESC key or clicking the **Cancel** button will close the box after the current step completes.



This tutorial continues with the following section, **Graph Updates**.

Graph Updates

If you have a calculation that takes a long time to build a dataset that you want to graph, you don't have to wait to display the entire graph. You can update the graph in segments.

The `RealTimeDisplay()` function in the "User Friendly Interface.c" file illustrates the technique.

This portion of the tutorial assumes that you have the "User Friendly Interface.c" file loaded and compiled from the previous example.

8. type the following in the **Script Window**:

   ```
   RealTimeDisplay()
   ```

   Press **ENTER**.

When you execute the `RealTimeDisplay` function, you are calling the following section from the C file (only partially displayed here; see the .C file for the full code). Comments are the green text with leading //.

User Friendly Interface.c

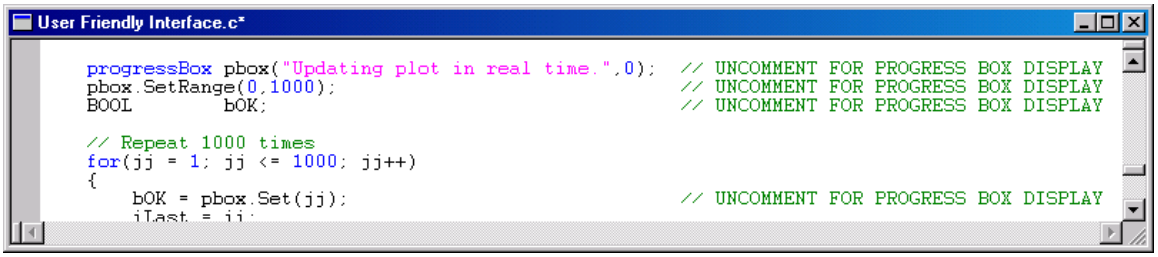
```
void RealTimeDisplay()
{
    Worksheet   wks;
    Dataset     dsX;
    Dataset     dsY;
    string      strWksName;
    string      strDSName1;
    string      strDSName2;
    GraphPage   graph;
    string      strGraphName;

    // Create a worksheet from the standard Origin worksheet template
    wks.Create("origin", CREATE_VISIBLE);
    // Get the worksheet name
    strWksName = wks.GetPage().GetName();
    // Attach to dataset and get column names
    dsX.Attach(strWksName,0);
    dsX.GetName(strDSName1);
    dsY.Attach(strWksName,1);
    dsY.GetName(strDSName2);

    // Create a graph from the Scatter graph template
    graph.Create("scatter",CREATE_VISIBLE);
    // and get its name
    Application.Pages().GetName(strGraphName);

    // Create a curve from the two dataset names
    Curve       cuvPlot(strDSName1, strDSName2);
    // and plot it
    graph.Layers(0).AddPlot(cuvPlot,IDM_PLOT_SCATTER);

    // Turn off Speed Mode, set scales, set point size and force graph to finish drawing
    LT_execute("layer.maxpts=0;x1=0;x2=10000;y1=0;y2=100;set %C -z 3;sec -p .1;");
```

A worksheet and a graph will be created and the graph will be updated in pieces. Note that there is also code which will display a progress box as well.

User Friendly Interface.c

```
//  progressBox pbox("Updating plot in real time.",0);  // UNCOMMENT FOR PROGRESS BOX DISPLAY
//  pbox.SetRange(0,1000);                              // UNCOMMENT FOR PROGRESS BOX DISPLAY
//  BOOL        bOK;                                    // UNCOMMENT FOR PROGRESS BOX DISPLAY

    // Repeat 1000 times
    for(jj = 1; jj <= 1000; jj++)
    {
//      bOK = pbox.Set(jj);                             // UNCOMMENT FOR PROGRESS BOX DISPLAY
        iLast = jj;
```

9. Return to the Code Builder interface and uncomment the four lines which are marked with "UNCOMMENT FOR PROGRESS BOX DISPLAY" (Remove the leading // from the 4 lines of code, but *do not* remove them from the four "UNCOMMENT FOR PROGRESS…" comments). Your code should now look like this:

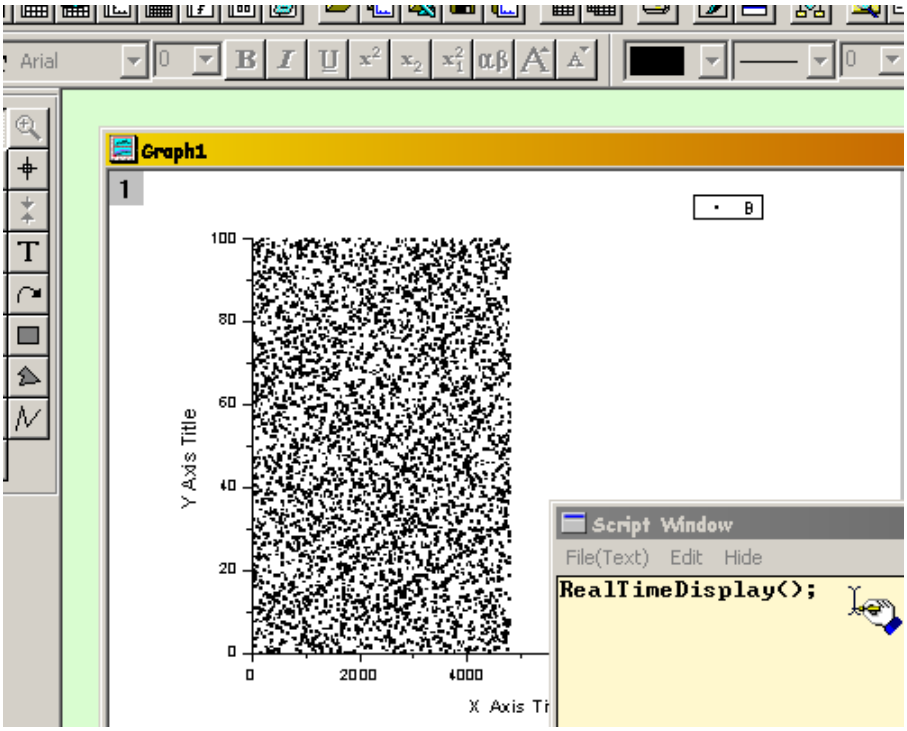


10. Click the **Re-build** button  .
11. Return to the Origin workspace and enter the following into the Script window:

```
RealTimeDisplay()
```

Press **ENTER**.

Note how the graph and the progress box seem to update simultaneously.



This concludes the **Programming Cues** tutorial.