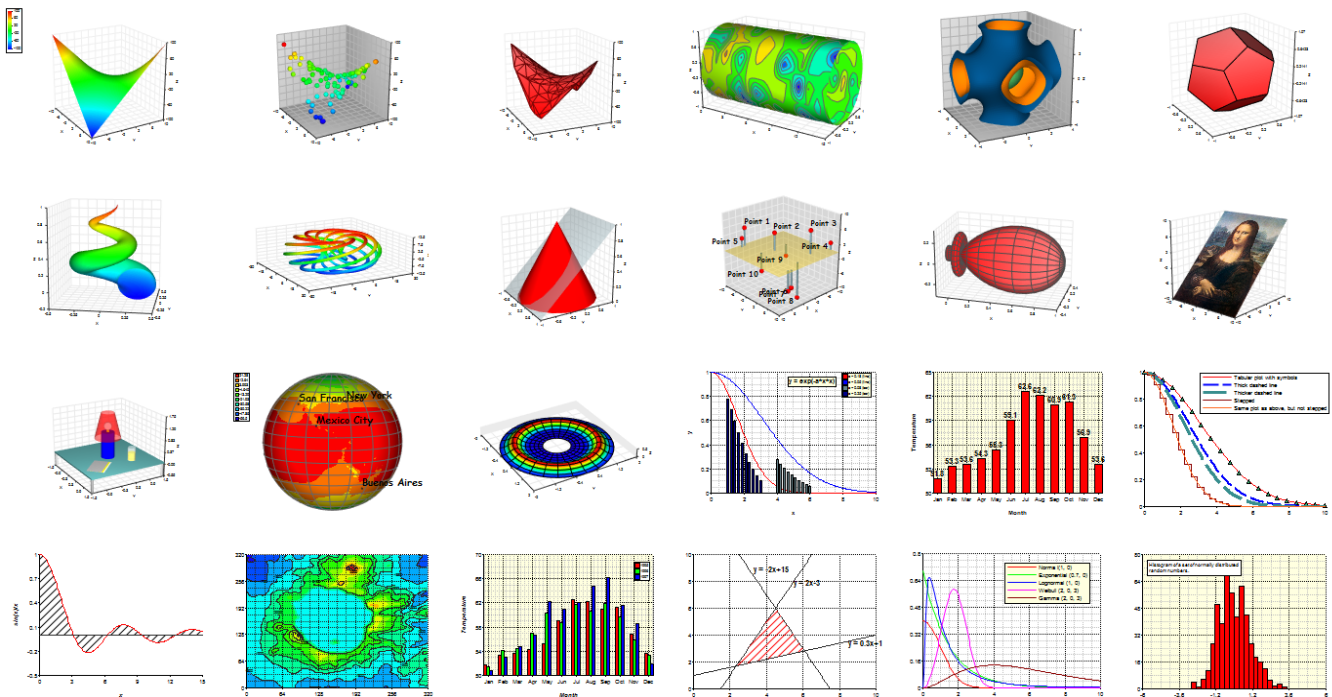


# TeraPlot – Graph Plotting and Data Analysis for Science and Engineering



TeraPlot graphing software gives you everything you need for publication quality graph plotting in science and engineering. At its core is the ability to create 2D and 3D graphs based on either tabular data or mathematical expressions. Single plot types can be used individually to create a basic graph, or multiple plots types can be combined to construct complex 3D visualizations comprising analytical expressions, tabular data, 3D objects, annotation, and image overlay. In addition to these primary features, a range of capabilities such as flexible data import and the ability to create graphs with just a few mouse clicks via graph wizards make TeraPlot a highly effective visualization tool. The program also provides curve fitting and data analysis features such as linear and nonlinear regression, histograms and box plots.

## Program Features

- Plot complex mathematical expressions in 2D and 3D using VBScript
- Base plots on a range of coordinate systems (e.g. cartesian, spherical, cylindrical) and combine them in the same graph.
- Import, store, and plot tabular data using internal TeraPlot spreadsheets.
- Create graphs with just a few mouse clicks using Graph Wizards.
- WYSIWYG multi-page, multiple graphs per page document layout. Easily add/remove/reorder document pages.
- Copy and paste graphs within and between TeraPlot documents.
- Export graph images to the clipboard or image files.
- Annotate pages with page captions, and individual graphs with graph captions, legends and colour keys.

- Complete flexibility to specify graph component colours, e.g. background, axes, axis labels etc.
- Combine any of the available 3D (2D) plots types in a single 3D (2D) graph.
- Rotate and tilt 3D graphs to any desired viewing angle.
- Animate 3D graph tilt/rotation parameters and optionally save animation to movie file.
- Scale relative lengths of x, y, z axes on 3D graphs.
- Highly flexible axis tick positioning and labelling options on both 2D and 3D graphs.
- Display graph walls and/or grid, or axes only.
- Specify 3D graph lighting directions.

## 3D Plot Types

- **Surface Plot - Regular Grid.** A 3D surface based on a regular grid, which can be cartesian, polar, cylindrical, or defined by parametric function coordinates (u, v). Options include colour mapping, texture mapping, transparency, 3D contour display, general 3D transformation, wire frame display.
- **Surface Plot - Irregular Data.** A 3D surface based on (x, y, z) data with arbitrarily positioned x, y values. Options include colour mapping, transparency, contour display, general 3D transformation, wire frame display.
- **Scatter Plot.** Draws a symbol (sphere, cube, cone) at each of a set of points in 3D space in a cartesian, polar, or cylindrical coordinate system. Up to two additional variables can optionally be associated with each scatter point, which can be visualised via the scatter point colour and/or size.
- **Isosurface Plot.** Draws surfaces of constant w for functions of the form  $w = f(x, y, z)$ , i.e. the 3d equivalent of contour plots. The data from which the isosurfaces are generated can be defined as a formula (analytical plot), or loaded from a file. For an analytical plot, the data is assumed to lie on a regular rectangular 3D grid. For data loaded from a file, the data can either lie on a regular grid, or arbitrary individual cells can be specified.
- **Bars - Regular Grid.** Draws a bar of height z at each of the plot's (x,y) points, which lie on a regular x, y grid.
- **Bars - Irregular Data.** Draws a bar of height z at each of the plot's (x,y) points. The points can be arbitrarily positioned in the (x, y) plane.
- **Series Plot - Regular Grid.** Joins the (x, y, z) points in a plot by straight lines to form a series of (x, z) or (y, z) subcurves. The option exists to additionally draw a vertical sheet at each subcurve to produce a "waterfall" plot. The points are constrained to lie on a rectangular grid with constant spacing in the x and y directions.
- **Series Plot - Irregular Data.** Joins the (x, y, z) points in a plot by straight lines to form a series of (x, z) or (y, z) curves. A vertical sheet can optionally be drawn at each curve to produce a "waterfall" plot. A given (x, z) curve can have arbitrary x values, but all its y points must be equal. A given (y, z) curve can have arbitrary y values, but all its x points must be equal.
- **3D Line.** Joins a set of points in 3D space by straight lines. The points can be defined in a cartesian, polar or cylindrical coordinate system.
- **Wireframe Plot.** Similar to surface plot, but draws wireframe only.

- **Object Plot.** Draws a set of 3D objects of predefined types which can be combined to create 3D scenes. Current object types include sphere, cube and cylinder, The objects are described by TeraPlot Object Description Language (GODL). The description can be loaded from a file or entered interactively.
- **Mesh Plot.** Draws a general 3D (e.g. finite element) mesh defined as a set of vertices and elements read in from a file with data formatted in TeraPlot native ascii format. Variables can be associated with the mesh vertices or elements, and the mesh colourmapped based on these variables. The option also exists to load dxf files (FACE data only).
- **Text Plot.** Draws a piece of text at each of a set of points in 3D space. The text plot is usually used along with other plot types to provide annotation at specific points in a graph. The points can be defined in a cartesian, polar (theta, phi or lat,lon), or cylindrical coordinate system.

## 2D Plot Types

- **Line Plot.** Joins a set of (x, y) points as straight line segments. The x values can be arbitrary or lie on a regularly spaced grid. Symbols can optionally be displayed at the points, and line styles such as thickness and dash pattern can be set. Shading underneath the line can be displayed over an arbitrary range relative to an arbitrary baseline. The option also exists to display the plot as stepping between the points rather than straight line segments. Because line plots can be analytical (i.e. based on mathematical expressions) as well as tabular, it's a simple matter with to plot a set of experimental points along with a formula they may correspond to.
- **Scatter Plot.** Places a symbol at each (x, y) point in the plot. Symbols can be arbitrarily scaled, and the fill colour and border colour set. The fill colour can be displayed as solid or patterned.
- **Bar Plot.** Draws a bar of height y at each of the plot's x points. The bar fill colour and border colour can be set, and the fill colour displayed as solid or patterned.
- **Contour Plot - Regular Grid.** Draws contours and/or colourmapped levels associated with a function of the form  $z = f(x,y)$  defined on a regular (x, y) grid. Contour labels can optionally be displayed, with a wide range of label configuration options such as font, scaling, orientation, and foreground and background colour. Because contour plots can be based on mathematical expressions, they can be used to draw implicit functions of the form  $f(x, y) = 0$ . For example, plotting  $xy - 5 = 0$ , would involve plotting  $z = xy - 5$ , and displaying the  $z = 0$  contour value. Families of such plots can easily be plotted by displaying multiple contours.
- **Contour Plot - Irregular Data.** Plots contours and/or colourmapped levels associated with a function of the form  $z = f(x,y)$  defined at arbitrary (x, y) points. All of the options described above for the regular grid contour plot are also available for this plot.
- **Line Series Plot.** Graphs a set of line plots, all of which share the same x values. This plot is drawn using the [Series/Category Graph Wizard](#). The individual lines plots in the line series plot have options identical to the line plot described above.
- **Clustered Bar Plot.** Created using using the [Series/Category Graph Wizard](#). Draws each series as a set of bars clustered around each x (category) value.
- **Stacked Bar Plot.** Created using using the [Series/Category Graph Wizard](#). Draws each series as a set of vertically stacked bars located at x (category) value.

- **Area Plot.** Created using using the Series/Category Graph Wizard. Draws a set of vertically stacked areas, each area corresponding to a series.
- **Text Plot.** Draws an arbitrary piece of text at each of the plot's (x, y) points. The text plot is usually used along with other plot types to provide annotation at specific points in a graph. Each plot point is defined by an x value, a y value, and a piece of text, and various options exist to set text font, scaling, and text alignment relative to the (x, y) location.
- **Error Bar Plot.** Draws an error bar at each of the plot's (x, y) points.

## Graph Wizards

Graph Wizards allow complete annotated graphs to be created with just a few mouse clicks. Once started, the graph wizard presents you with a set of input stages, the first of which is choice of plot type. This is followed by a data entry stage, which is then followed by one or more additional graph parameters stages. When all data has been supplied, a preview of the graph is displayed, and this can either be accepted, or previous stages can be returned to to make further enhancements. If the final result is accepted, the new graph is created in a new TeraPlot project window. Three graph wizards are currently available:

- The **Series/Category Graph Wizard** automates graph creation with data which takes the form of a collection of series plotted across a set of categories, with these terms having exactly the same meaning as they do in Excel when referring to data plots. An example might be a requirement to plot the 12 average monthly temperatures for six different years, giving  $6 \times 12 = 72$  data points. In this case each set of 12 points representing the data for a given year would constitute a series (giving 6 series in total), and the months themselves would represent the categories (giving 12 categories). In this case the number of series was 6, but it can be as small as 1 or as large as hundreds.
- The **XYZ Graph Wizard** automates graph creation based on data consisting of a list of (x, y, z) values. Typically, the x,y values of the points entered in this graph wizard will be arbitrary, in that they won't lie on a regular grid of any sort. Available plot types are: Surface, Scatter, Bar, Line, and Contour. Each plot type has its own dedicated parameters page in the graph wizard, allowing for a high degree of tailoring before previewing and creating the graph.
- The **Financial Data Graph Wizard** automates graph creation based on financial data. Available plot types include High-Low-Close (HLC) , HLC-Volume, Open-High-Low-Close (OHLC), OHLC-Volume, OHLC Candlestick, and OHLC Candlestick-Volume. Data is presented to the wizard as a set of columns, with the first column representing the x values. Depending on the plot type being created, this should then be followed by up to a further 3-5 columns representing opens, highs, lows etc. The main plot types are 2D, but the option exists to plot HLC or OHLC data as 3D plots with the x values varying along the 3D graph x axis and the HLC or OHLC data varying along the 3D graph y axis.

## Statistics/Data Analysis

TeraPlot provides data analysis capabilities in the form of statistical functions and standard statistical analysis plots such as histograms, box plots, probability plots, and linear and nonlinear regression.

- **Statistical Functions.** Statistical distribution plots can be used to plot statistical distributions (PDF or CDF) based on supplied distribution parameters. Available distributions include: Normal, Exponential, Lognormal, Weibull, Gamma.

- **Statistical Plots.** Statistical plots include: linear regression, probability plot, PPCC plot, and box plot
- **Nonlinear Regression.** TeraPlot provides a multidimensional nonlinear regression facility which can be used to fit data to a general functional form using the Levenberg- Marquardt Method.

## Program Automation

Automation provides the ability to remotely start TeraPlot via a small driver program written in e.g. VB.Net or C#. Much of the menu-driven functionality which would normally be accessed manually can then be accessed and manipulated programmatically. Automation is therefore potentially useful in situations where you find yourself repeatedly performing the same manual tasks in TeraPlot.

As an example, imagine you are required to create 6 surface plots laid out on a page in three rows of two, and that you may even be required to create multiple batches of such collections of plots. Rather than manually starting TeraPlot and performing all of the mouse clicks required to create six graphs, each containing a surface, you could write a short program which created the graphs automatically for you. Your program would first start TeraPlot and create a new TeraPlot document. Then, inside a loop which executed six times, you could create a new graph, positioned appropriately on the page, to which you would add a surface. If the surface was an analytical surface, it would be defined by a text string containing one or more expressions defining the surface. If the surface was tabular, it would be defined by an array of numbers representing its z values. Once all of the graphs had been created, the document could be programmatically saved and TeraPlot closed.

The main program constructs by which these steps are achieved are the objects provided in the **TeraPlot Object Model**. For example, in the case described above, TeraPlot would be started using the **TApplication** object. The document and page functionality would be accessed using the **TDocument** and **TPage** objects. Finally, the graph and surface functionality would be accessed using the **T3DGraph** and **T3DSurfacePlot** objects.

This is just one example of the power that automation brings to TeraPlot. Controlling the program from a programming language provides a wide range of possibilities, from writing code which interactively controls TeraPlot via your own dialogs, to creating movies utilising any combination of TeraPlot features you wish.