

Requirements for graphing

Goal:

Provide uniform interface for representing 1D, 2D and 3D scientific graphics.

Requirements:

Low-level tools of plotting: [Applications may not see this level.](#)

A uniform interface to draw the 2D points (markers), lines (rectangle), arc, cubic curve, quadratic curve, and text in different graphics context.

A uniform interface to draw the 3D points (markers), lines (rectangle), arc, curve, vertex, text and 3D objects in different graphics context.

Middle -level tools of plotting:

1D plotting: (one or multiple) x – (one or multiple) y plotting
(one or multiple) r – (one or multiple) θ plotting
Line, scatter, and stack plotting

2D plotting:

Color-mapped 2D images.
Provide several predefined color-map
Support user defined color-map
Contour lines

3D plotting:

(one or multiple) x – (one or multiple) y - (one or multiple) z plotting
(one or multiple) r – (one or multiple) θ - (one or multiple) φ plotting
Iso-surface plotting
Basic 3D objects (sphere, cylinder, ...)
Plot molecules

Axis, title, legend and label for the plotting

Labels for points, lines, etc.

Grids

Guide lines (e.g., $y=15$), [Guide functions as well, such as \$\sin\(x\)\$](#)

Uncertainty for on data points ($x/y/z$ symmetric/asymmetric).

Confidence intervals for theory functions

Support the overlay function. The plots or the groups can be in the different layers

within the same canvas. [Need to control order of objects on the plot](#)

The plots can be grouped, and user can change the group properties.

Application developers must not have direct control of representation --- colours, symbols, line styles, fonts, etc. --- but must instead indicate whether lines are related, contrasting or unrelated to other lines.

High -level tools of plotting:

- Application defined plottable objects
- Show the related information when the user view the plotting, such as show the value of the given point
- Adjust axis ranges
- Adjust colour map properties (floor, ceiling, map choice) to highlight data features
- Move canvas objects on plot to adjust application parameters
- Select points, lines, ranges, regions
- User defined view transformations, such as the log and linear scaling of the data
- Application defined view transformations such as divide by Fresnel reflectivity, multiply by Q^{**4} , compute spin asymmetry, Q_x - Q_z vs. pixel representation
- Application defined context menus for plot objects
- Hide/show plot objects
- Raise/lower plot objects
- Application defined operations on plot canvas (e.g., show average vs. show individual lines)
- Application defined operations on lines

Publication graphics:*Not sure this is a requirement*

- All the properties of the low-level tools and high-level tools can be interactively changed
- Update the plotting in real time in the native window system (Windows, Mac, X-Window) when the properties are changed
- Page layout tools for graphs
- Copy high level graph objects onto page layout graphs
- Operations, such as the rotation, transformation copy, paste, dragging, of plotting objects.
- Ability to see and change values associated with plot objects

Output of plotting:

- Support the common image formats, such as gif, jpg, png, tiff
- Support PDF, PS and SVG format
- Paste plots and/or data in spreadsheet/document editor.

Document and Deployment:

- Well documented, tutorial materials and examples
- Easily to be deployed and installed
- Independent on the platform, should work on Windows, Unix/Linux, and Max OS