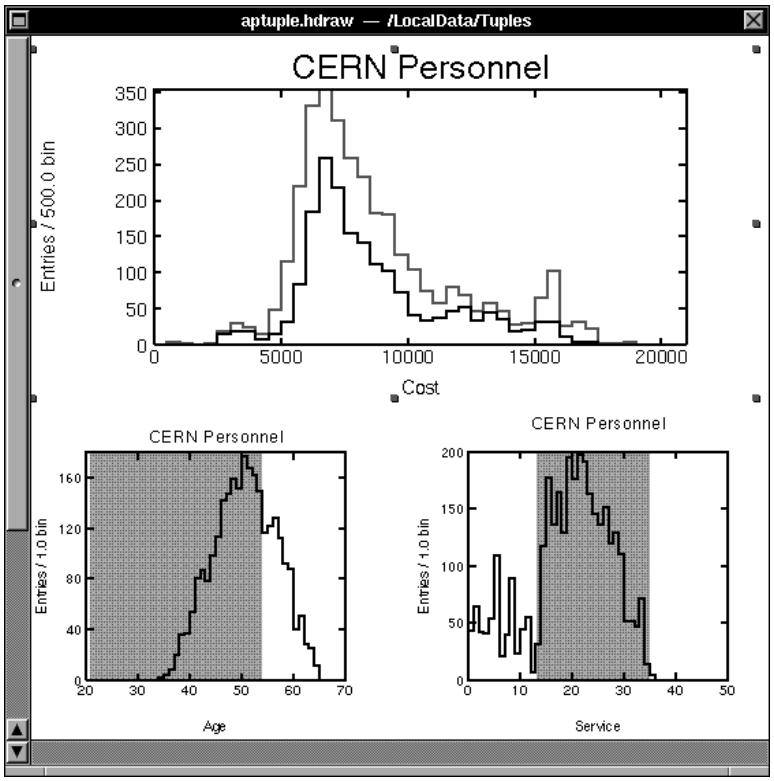
HippoDraw

Users Manual and Reference Manual (Release 1.1)

Mike F. Gravina
Paul F. Kunz
Paul Rensing

Stanford Linear Accelerator Center
Stanford University
Stanford CA 94309



Disclaimer Notice

The items furnished herewith were developed under the sponsorship of the U.S. Government. Neither the U.S., nor the U.S. D.O.E., nor the Leland Stanford Junior University, nor their employees, makes any warranty, express or implied, or assumes any liability or responsibility for accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use will not infringe privately-owned rights. Mention of any product, its manufacturer, or suppliers shall not, nor is it intended to, imply approval, disapproval, or fitness for any particular use. The U.S. and the University at all times retain the right to use and disseminate the furnished items for any purpose whatsoever. Notice 91 02 01

Copyright 1992
by
The Board of Trustees of the
Leland Stanford Junior University.
All rights reserved.

Work supported by the U.S. Department of
Energy under contract DE-AC03-76SF00515.

Table of Contents

1. Introduction 1

2. A Tutorial 2

3. Known Limitations and “bugs” 5

4. Reference Manual 6
   - 4.1 Info 6
   - 4.2 Document 6
   - 4.3 Edit 8
   - 4.4 Graphics 8
   - 4.5 Fonts 9
   - 4.6 Tools 9
     - 4.6.1The Tools Panel 10
     - 4.6.2The Graphics Inspector 11
     - 4.6.3The Tuple Inspector 11
     - 4.6.4 Tuple List Inspector 12
     - 4.6.5Data Selection Inspector 13
     - 4.6.6Graph Options Inspector 14
     - 4.6.7Axes Selection 16
     - 4.6.8Plot Statistics Inspector 17
     - 4.6.9Cut Options Inspector 18
   - 4.7 The Remaining main menu items. 20

5. Programmer’s Guide 21
   - 5.1 The application directory: HippoDrawSrc 21
   - 5.2 HippoDraw Sources: Hippo.subproj 23
     - 5.2.1Unmodified Draw Sources 23
     - 5.2.2Modified Draw Sources 24
     - 5.2.3New HippoDraw Sources 24
   - 5.3 Building Custom Application with HippoDraw 26
   - 5.4 Changes in the 1.1 Release 27

1. Introduction

HippoDraw is a combination of the Hippoplotamus (or *hippo* for short) histogram and plotting package and the NeXTstep demonstration application Draw. Hippo's input is a table of numbers with probably a small number of columns but perhaps with large number of rows. This data set is commonly sometimes called an n-tuple. Hippo manages the data set and can perform frequency distributions in the form of histograms or density plots using color or grey scale. To the original Draw application, a graphic object called a Plot was added. This object can be manipulated like the other graphic objects. Although the plots appear in the canvas as drawing objects, they remain connected to the n-tuple data. Thus one can change any of the display characteristics that are supported by *hippo*.

This manual is organized as 3 sections. The first is the introduction you are currently reading. In section 2 we have a simple tutorial to help the reader get started. The next section is a reference manual to all the facilities of HippoDraw that are not in the base Draw application. The last section describes how to incorporate HippoDraw within another application.

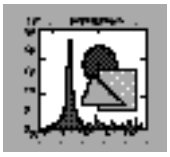
HippoDraw reads n-tuple data in the *hippo* format. Hippo files can be either in binary or plain text form. Information on the format of these files, and how to create them can be found in the Hippoplotamus Users Guide and will not be repeated in this manual. A simple example of the plain text form can be found in the on-line help document.

2. A Tutorial

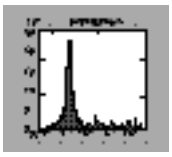
This section contains a short tutorial on the use of HippoDraw. It is assumed that the reader is familiar with NeXTstep applications and no attempt will be made to explain things that are standard in the NeXTstep environment.

One launches HippoDraw in the usual way by double clicking on the application Icon

which gives you an empty drawing document; or by double clicking on a HippoDraw document file icon.



which opens a previously saved document; or a *hippo* binary file icon



which gives you an empty drawing document and imports the n-tuple in the manner described below. Another way to launch HippoDraw is via the Services menu item from another application in which the plain text format of a *hippo* file can be selected.

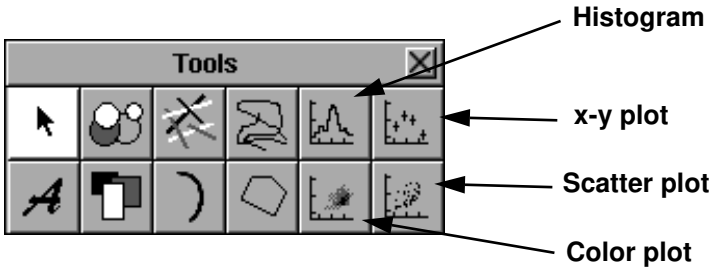
By the way, the peak shown in each of these icons was produced by HippoDraw. The graphics in the document icon comes from the motif of the Draw application. For this tutorial, we'll assume you launched the application by double clicking on the application icon and are now faced with an empty draw-

ing canvas. This canvas, called the "document" is just like the one in the Draw application, except the grid has been turned on by default.

The next step is to import an n-tuple file that will be used as the data input to the plots. There are two ways to do this. First, there are two submenu items under the **Document** main menu, as shown to the left, titled **Import tuple** and **Import text tuple**. The first menu item is used to open a *hippo* file that is in binary format (*e.g. one* one with the .hippo file name suffix), while the second is used to open a *hippo* file that is in plain text format. An alternate method for opening a binary *hippo* file is to drag the file icon over the document window. Try opening the aptuple.hippo data set that came with this distribution. The attributes of the impoted n-tuple will be displayed in the **Tuple List** inspector panel.

When you import an n-tuple via one of the two **Import** menu items, the open panel allows you to import the n-tuple *by reference* or *by copy*. This selection has no effect until the drawing document is saved. If you imported by reference, then only the path to the n-tuple file is saved in the drawing document. If you imported by copy, then a copy of the n-tuple is saved with the drawing document. It is recommended that you use the former, which is the default, as your n-tuple files could be quite large, and you may want several drawing documents sharing the same file. If you drag a *hippo* binary file icon over the document window to import the n-tuple, then a panel will appear to give you the by-reference or by-copy choice. Likewise, if you double click on a the hippo file icon, you get this panel.

The next step, which could be easily missed if you weren't reading this tutorial, is to generate a default plot. This is done via the **Tools** panel as shown below. It appears that four kinds of



Plot objects can be created, but these buttons only control the initial configuration of one kind of Plot object and it can be

changed later. To create a default 1D plot, *e.g.* a histogram, click on the button that looks like one. To create a 2D plot, click on the appropriate button. The 2D color plot comes up in grey scale initially, but you can change it via the **Graph Options** inspector panel. Try the 1D plot button first. To change which n-tuple column is used for the axes of your plots, use the **Data Selection** inspector panel. To change the axes scale or the binning, use the **Axes Selection** inspector panel. You reach these inspector panels by clicking on the pop-up list button in the **Tuple Inspector** panel.

The plot is placed in the upper left corner if that area is not occupied by another graphic object. The next plot is placed to the right of the first, if the space is free. Each additional plot is placed below the existing graphics. When the page fills up, a new page is generated automatically.

Plots are like all other graphic objects of the drawing program. They can be re-sized, moved, copied, cut, and pasted. You can copy a plot and paste the Encapsulated PostScript into another application. You can use the **Graphics** tool panel to colorize your plot, change the line size, etc. However, with this release, the text in the plot is treated as part of the plot, not a separate graphic object. You can turn off this text and overlay the plot with a text graphic object if you wish.

A tricky part of HippoDraw is applying cuts to the data. The technique is to select a plot to which you wish to apply a cut. Then use the Cut Options inspector panel to first create a new cut (and its corresponding plot). One can also first create a cut without selecting a plot and apply any cut or cuts to any plot or plots, selectively removed them from some plots, but not others, etc. It may take a little practice to see how this works. There is one rule that is always followed: *you can not apply a cut to a plot without the variable being used by the cut being displayed in its own plot.*

The reference manual section of this document gives details on how the plotting and n-tuple aspects of HippoDraw work. However, the pure drawing aspects, which were inherited from the original Draw application are not documented here. Only the original Draw **Help** menu item is available for that part of HippoDraw.

3. Known Limitations and “bugs”

There are a number of known limitations in this release of HippoDraw. Some of them might be considered “bugs”. However, in the interest of giving users an opportunity to try out what the authors consider HippoDraw’s unique innovations, the release has been made in spite of the limitations and known bugs. The known limitations are listed below…

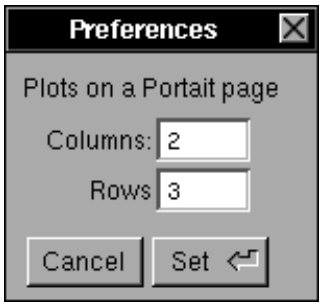
- The text in the plots are not separate text graphic objects.
- If a *hippo* file contains displays, they are ignored.
- The x-y plots aren’t as pretty or refined as other applications on the NeXT.
- There’s no way to use a 2D histogram to define cuts.
- The statistics inspector only gives counts; no means or variances are available yet.

It is planned to continue development of HippoDraw and the most serious of these limitations will be fixed soon. It is first important to evaluate the usefulness of its unique features. Also with the introduction of new features of NeXTstep 3.0, it may be possible to incorporate HippoDraw’s plots into any application, such as FrameMaker, Diagram!, etc.

4. Reference Manual

4.1 Info

The **Info** main menu item has four sub menu items which are shown to the left. The **Info Panel** item is a standard NeXT info panel listing the authors of the program. Please read it. The **Plot Preferences** brings up a panel for setting the default size of plots. This panel is shown below. Plots are layed down on a grid



and one sets the default number of columns and rows with this panel. HippoDraw will calculated the default plot size. The **Help** item brings up a help document which is the only documentation available for the Draw part of the application. Finally, the **Legal** menu item brings you important information from our sponsors.

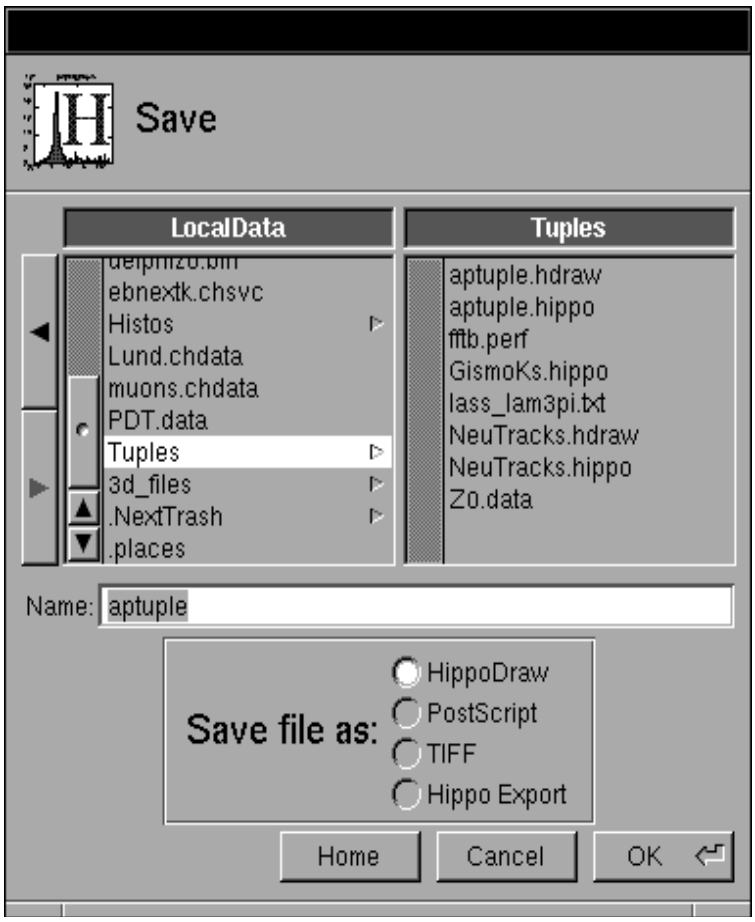
4.2 Document

The **Document** main menu item controls the opening and saving of HippoDraw documents and import of the n-tuple files. The submenu is shown to the left.

The **Open** menu item opens a HippoDraw document file (.hdraw) which is a file for the drawing information and either a reference to a n-tuple file or the n-tuple itself. The **Import tuple** menu item opens a binary n-tuple file, while the **Import text tuple** menu item opens a n-tuple file formatted in plain text. In either case, the Open Panel that appears has radio buttons to allow one to open the n-tuple file by reference or by copy. The only difference will be when the document is saved. If the de-

fault import by reference is chosen, then only the path to the n-tuple file will be written to the document file. If import by copy is chosen, then a copy of the n-tuple data will be written with the document file, making it much larger, of course. The **New** menu item opens a new blank document canvas.

There a number of options for saving the document. The **Save** menu item saves the drawing and n-tuple reference or a copy of the n-tuple as a HippoDraw document file. The **Save As** menu item is the same as **Save** but allows one to change the file name. It is automatically invoked when the document has not yet been saved as with many other NeXTstep applications. The **Save To** menu item allows one to save the document in different formats. The **Save To** save panel is shown in below.



The default option is **HippoDraw** which is the same as **Save** and **Save As**. The **PostScript** option saves the whole document as an EPS file, while the **TIFF** option saves it as a TIFF file. The **Hippo Export** option saves the opened n-tuples as a *hippo* binary file (.hippo). However, this option is only useful if the hippo data was imported by copy.

The **Save All** menu item saves all the open documents. The **Revert to Save** menu item discards the current document and re-opens the originial. The **Close** menu item closes the current document.

4.3 Edit

The **Edit** main menu item is pretty much a standard NeXTstep **Edit** menu as shown to the left. Using **Cut**, **Copy**, or **Paste** plots just like other drawing objects. Note that you can not spell check the text generated by *hippo* in the plots.

4.4 Graphics

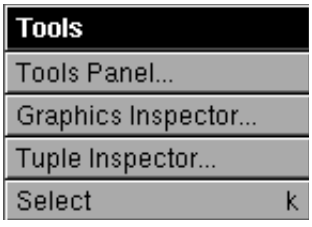
This **Graphics** main menu item (which was called **Format** in the original Draw application) controls the placement of selected graphics objects on the canvas. Note that HippoDraw disables menu items that are not valid at any moment. Disabled menu items appear with grey text. Most of the items are pretty obvious and have names similar to other drawing programs. The **Bring to Front** and **Send to Back** menu items bring the selected graphic in front of or in back of other graphic objects it might over lay respectively. The **Group** menu item takes multiple selected graphics and groups them into one compound graphic. **Ungroup** undoes the effect of **Group**. **Lock** prevents the selected graphic from being selected, resized or moved. **Unlock** undoes the effect of lock. **Natural Size** will reset a graphic to its natural proportions if meaningful. Thus a rectangle will become a square, an ellipse a circle, etc. Any imported EPS or TIFF files will reset themselves to their original size. A *hippo* plot will reset itself to the default size. The **Grid** menu item brings up a submenu for controlling the back grid and the snap to grid features. Unlike the Draw application, the grid and snap to grid is turned on by default in HippoDraw.

A special submenu is **Plots** as shown to the left. These are special alignment tools dealing with plots. These items align one plot with another. To use, first select the plot to which you want to align other plots. Then select one or more plots (shift-click). The **Overlay** menu item will resize and move the second plot so that it overlays with the first. The titles and axes labels are turned off for all but the first plot when **Overlay** is used. Overlayed plots will keep their aligned axes. You can over lay histograms with different bin widths and the y axes will be adjusted correctly. To un-do an overlay, use the **Unoverlay** menu item. The **X Range** menu item will change the range displayed on the x-axis of the second plot so that is the same as the first plot. Similarly, the **Y Range** menu item will change the range displayed on the y-axis. The **X Bins** and **Y Bins** menu items will align the second plot to have the same number of bins on the x and y axis, respectively, of the first plot.

Getting back to the **Graphics** menu, the remainder of the menu items should be obvious. The **Add Page** item is a feature of HippoDraw not found in the original Draw application. One word of caution when using the **Page Layout** menu. If you have graphic objects near the top of the page in portrait mode and then switch to landscape mode, you'll find that these objects may only be partially visible. This *feature* is inherited from the original Draw application.

4.5 Fonts

The **Fonts** main menu item is a very standard NeXTstep Fonts menu item and will not be described in this manual. In the current implementation, only the font of text created with the text tool can be changed and not the font in the plots created by *hippo*. If you want to put in different fonts for titles or labels, then use the **Graphics Inspector** described in 4.6.2 to turn off the title and axes labels and use the text tool to put in your own labels.



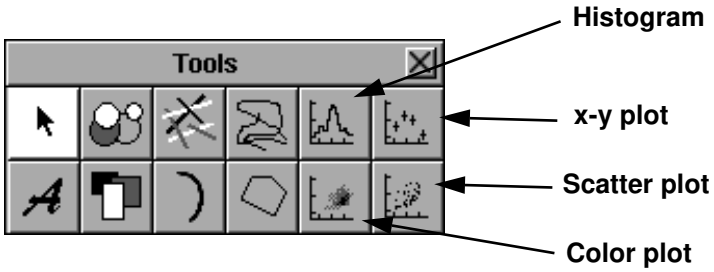
4.6 Tools

The **Tools** main menu item brings up the various graphic tools and inspector panels. It's shown to the left. Its purpose is to be able to bring up the graphic tools panel and other inspec-

tors to change the attributes of graphic objects and plots. The **Tools Panel...** brings up the panel to select what type of graphic will be drawn next. The **Graphics Inspector** brings up a panel to change various attributes of selected graphics. The **Tuple Inspector** brings up the inspector panel for plots and n-tuples. The **Select** menu item returns the cursor to a selection tool.

4.6.1 The Tools Panel

The **Tools Panel** looks is shown below. The 8 left most



drawing tools are the same as the original Draw application. The four right most tools are for creating new plots bound to the currently selected n-tuple. These buttons control only the initial form of display and can be changed later. The histogram button creates a plot which is a projection of one column. The color plot button projects two columns and displays them as a grey scale density plot. It can be changed to a color scale later. The x-y plot button takes one column as the x-axis and another as the y-axis and displays all rows as rectangles. The scatter plot button is similar to the x-y plot but displays all rows as a very small point and is intended to be used when the number of rows is vary large.

Plots are inserted into the drawing canvas starting in the upper left corner if it is not occupied by a graphic object, then to the right if space is free, and finally below current graphics. If no space is available on a page, then a new page is automatically created. If one is creating a new plot which is the same type as the currently selected plot, then the new plot will be a copy of the selected plot including its size. If no plot is selected when a new plot is created, then the default plot size is created. See the **Plot Preferences** menu item on page 6 for changing the default plot size.

4.6.2 The Graphics Inspector

The **Graphics Inspector** panel is the same as the one in the Draw application. However, for the moment, the text in the plot is considered part of the line characteristics.

4.6.3 The Tuple Inspector

The **Tuple Inspector** panel controls the options of the Hippo displays in the plot objects. It displays attributes of a n-tuple or plot by grouping them in six sub-panels. One can select the sub-panel with the popup list button at the top. The **Tuple Inspector** shows the attributes of the current selected plot, or the first plot selected, if multiple plots are selected. To select a plot, just click on it. To extend your selection to multiple plots, shift click on the additional plots.

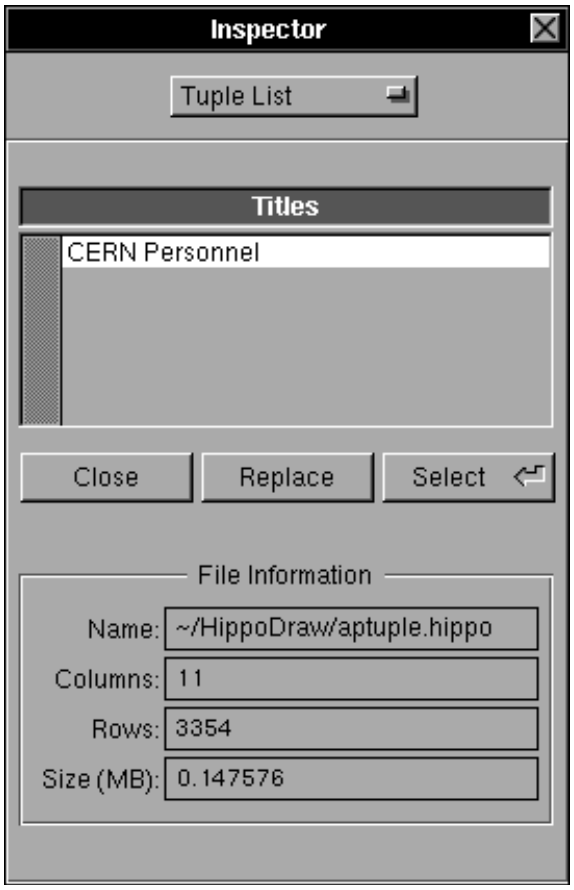
The six sub-panels, or inspectors, are briefly described below…

- **Tuple List** Inspector - list each of the opened n-tuples files.
- **Data Selection** inspector - lists the selected n-tuple's column labels and selects which column is bound to which display axis.
- **Graphic Options** inspector - controls various options of the display including the title and labels.
- **Axes Selection** inspector - controls the axes scales and number of bins for histograms.
- **Plot Statistics** inspector - displays some statistics of the display.
- **Cut Options** inspector - displays and controls cuts.

Each of these inspectors will be described in more detail in the following sections.

4.6.4 Tuple List Inspector

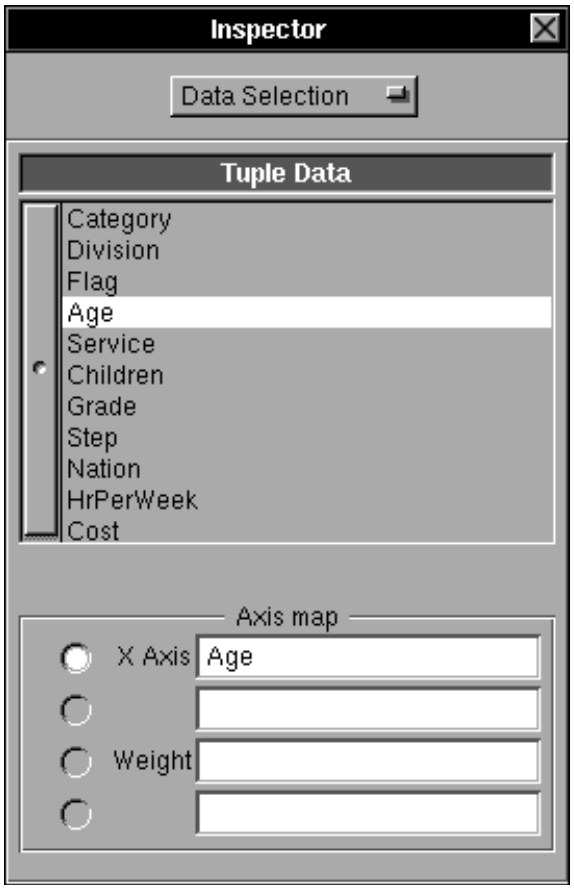
The **Tuple List** sub-panel, is shown below. In the browser it shows the titles of the currently opened n-tuples or "<none>" if it doesn't have one. This inspector is automatically brought up when an n-tuple is opened. The **Select** button is used to change which n-tuple is used for the next plot created with the plot tool. The **Replace** button is used to change which n-tuple the currently selected plot will use for its data. Thus, one could compare data sets by opening each n-tuple data set, create the plots to display the data, and replace the n-tuple used by the plot. The Close button will close the n-tuple file and free up the memory used by it. However, you can not close a file that is currently being used by a plot.



The **File Information** box displays more information on the selected n-tuple. The **Name** box displays the full path name of the n-tuple file, or the relative path from the user's home directory. If the path is too long for its full name to be visible, then you can select the text and scroll it with the cursor keys. The number of columns and rows of the n-tuple are displayed in the two boxes below the file name. Finally, the size of the n-tuple file, calculated from the number of rows and columns, is shown in its box.

4.6.5 Data Selection Inspector

The **Data Selection** sub-panel is shown on the below. It displays a browser with the labels of the columns of the n-tuple of the currently selected plot. This panel is automatically brought up when you create a new plot. One uses this panel to



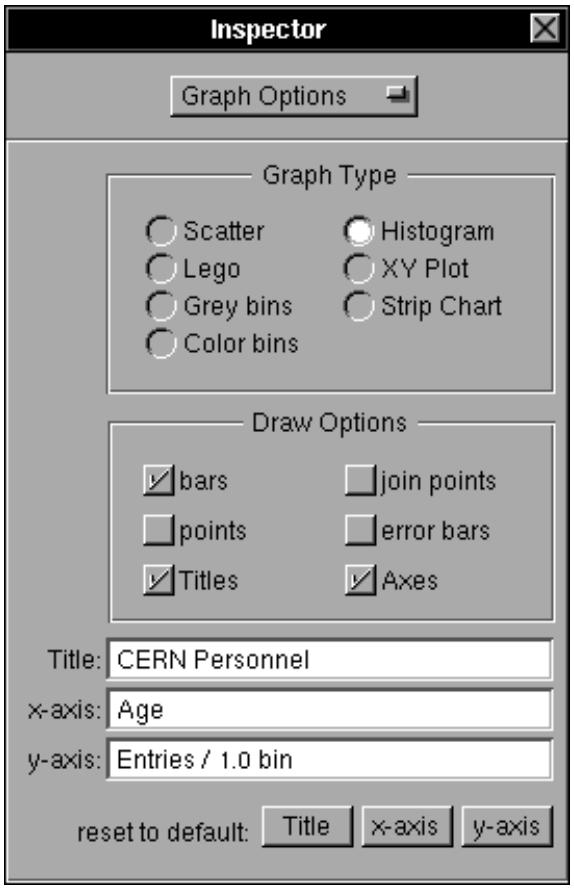
select the column to be used for each axis of the display via the buttons in the **Axis map** box. First click on the radio button next to the form field one wishes to change, then click on the column label in the browser. Note that the **Axis map** will display titles that are appropriate to the type of plot being displayed. That is, the map for a 1D histogram plot will be different from that of a 2D histogram or x-y plot.

For histograms, either of one dimension or two, one can weight the entries with data from one of the columns. first click on the button next to the weight form field, then click on the column to be used as weight. To remove a weight, click on the column to be used on the weight again; it toggles on and off.

4.6.6 Graph Options Inspector

The **Graph Options** sub-panel is show on the next page. The upper box controls the type of plot to be generated. No matter which tool was used to create the plot, you can change it by clicking on one of the radio buttons.

A Scatter plot is one where a pair of x-y points is plotted as a single point. It would normally be used when one has a data set with a large number of rows. The Lego plot has not been implemented yet. A plot with Grey bins is a 2D histogram with a grey scale representing the number of entries in each bin. White is used for an empty bin and black is used for the bin with the most number of entries. A linear relationship in greyness and the number of entries is used for the remaining bins. A plot with Color bins is similar to one with Grey bins but the color spectrum is used instead. A bin colored black has no entries in it, and a bright purple is used for the bin with the most entries. An x-y plot is normally used when there is a small number of rows. The option to join points makes sense for this kind of plot. It also has the options of selecting which columns correspond to the error on the x and y axis via the **Data Selection** sub-panel. A histogram is a projection of one column. It is really an x-y plot in which *hippo* calculates the y-axis by accumulation into bins. A Strip Chart type of plot is similar to an x-y plot except that the x-axis is assumed to be an ordered list. The plot will search the n-tuple for the smallest value on the x axis, then start plotting from there to the end of the n-tuple, and then wrap around to the first entry.



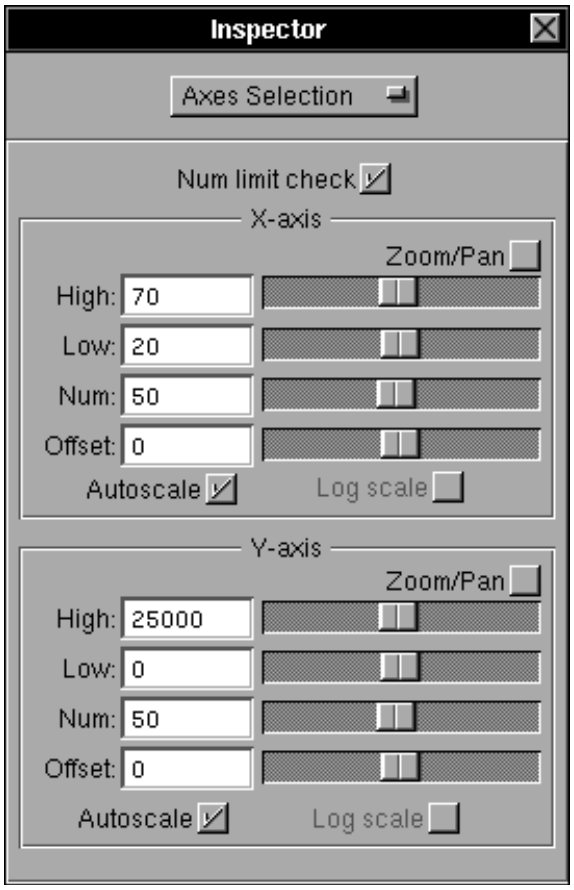
For some of the types of graphs, the lower box contains further options. HippoDraw will only respond to options are meaningful to the selected plot(s). You can also turn off the drawing of the title and labels of a selected plot. This feature is useful if you want to overlay plots in which case only one plot would have its titles and labels turn on.

The lower part of the panel allows you to change the title and labels. The default values are taken from the n-tuple data set. After changing a entry, you can restore the original by clicking on one of the three lower buttons. Note that for histograms, the bin width is calculated and becomes part of the axes label.

4.6.7 Axes Selection

The **Axis Selection** sub-panel controls the range displayed in the plots. it is shown below.One can change the range of data being displayed by either dragging on the sliders, or typing in the desired range in the form fields. The number of bins is changed in the same way, and it applies to the number of bins within the display range. The offset slider and form fields move the edges of the bins by a percentage of width of the bins. For some 2Dplots, the number of bins and offset controls have no meaning and are disabled. For 1D histograms, the high and low range that is displayed on the y-axis can be adjusted.

The Zoom/Pan check box changes the actions of the high and low sliders. When checked, the high slider turns into a zoom slider which changes but the high and low range in op-



posite directions. The low slider will turn into a pan slider which changes the high and low range in the same directions.

These sliders are not your ordinary NeXTstep sliders. We've added a special feature that you'll find useful for fine control of your display. If you hold down the Alternate key while clicking and dragging on these sliders, then the amount of change of the ranges will be reduced by a large scale factor.

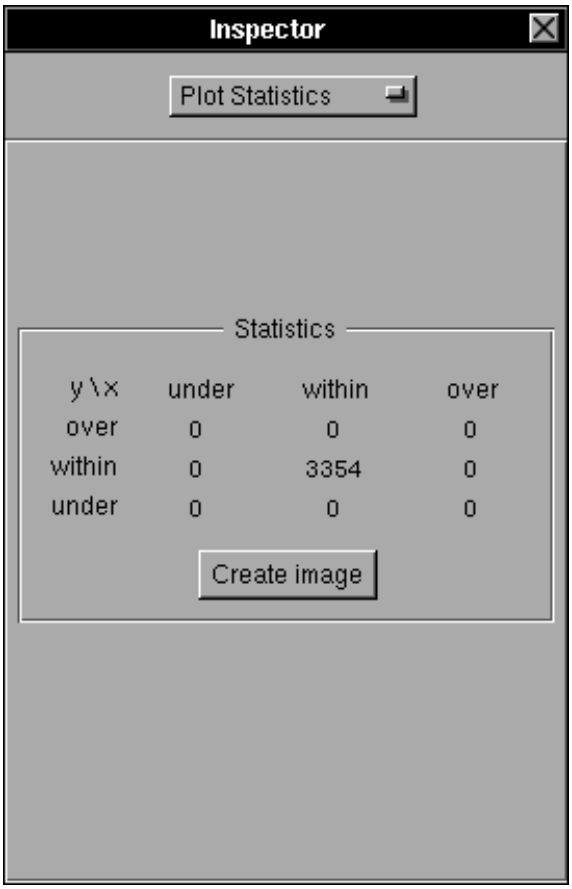
By default, the range displayed is automatically changed to show all the data. One can turn off this feature with one of the **Autoscale** check button. The autoscale of x- or y-axes are controlled separately. The appropriate button is automatically turned off if the range is adjusted with the sliders or by typing in the form fields. When applicable, the scale on either axes can be made logarithmic or linear with the **Log scale** check buttons. The button is disabled when a 1D or 2D histogram is selected.

It is easily possible via the slider or the form fields to request a very large number of bins which would take a long time to plot, or exceed the available virtual memory space. To help prevent such "accidents", the number of bins is by default limited to 500 on each axis. If you attempt to exceed this limit, an alert panel will appear, giving you a choice to continue or take the default upper limit. One can turn off the limit with the **Num Limit Check** check box near the top of the sub-panel.

4.6.8 Plot Statistics Inspector

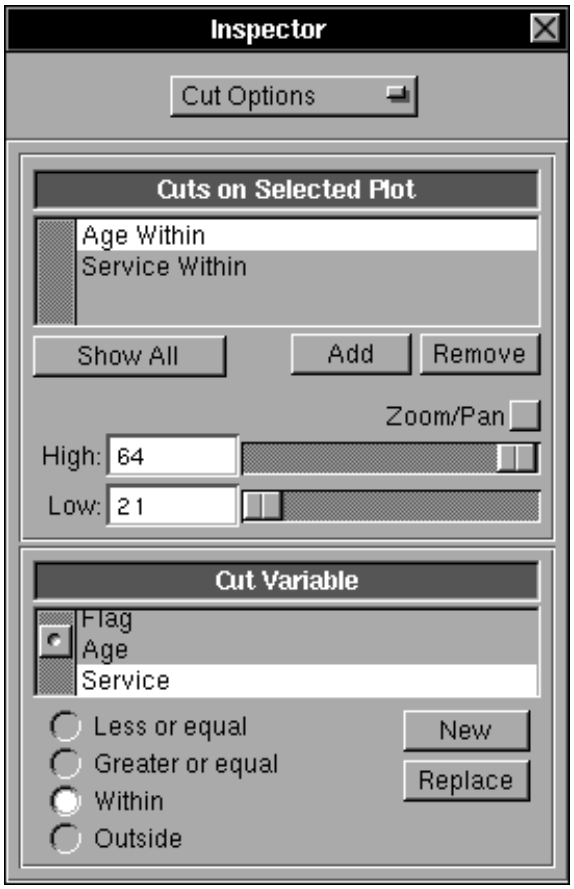
The **Plot Statistics Inspector** panel is shown on the next page. It shows the number of data points within the visible display of the plot as well as the number of data points under and over the displayed range. If the selected plot is a weighted histogram, then the sum of the weights is displayed. The **Create image** button will create an image of these statistics and place it into the drawing canvas.

With this release, the **Plot Statistics** inspector has limited utilty. In the future we hope to add means and variance. We also want to continously update the image created in the canvas, should the plot attributes change.



4.6.9 Cut Options Inspector

To further understand the data, it is frequently useful to allow data points from one column of the n-tuple to be included in one plot only if the data in another column is within a certain range. The **Cut Options Inspector** controls cutting of the data. Its panel is shown on the next page. The panel has two main parts. The lower part is for creating cuts while the upper is for applying them to existing plots. To create a new cut, one would select the column to be used as the cutting variable in the mini-browser of the n-tuple, then click on the **New** button. A new plot will appear on the drawing canvas with a shading to indicate the included region. Then you would apply the cut to existing plots. A short cut is to first select a plot by clicking on it. The labels of



the n-tuple columns used by that plot is display. One would then create a new cut with the **New** button and the cut will be applied at the same time. One of the radio buttons must be also selected. You can change the cut range with the sliders in this inspector.

In the upper part of the panel, a browser usually shows which cuts are being applied to the selected plot, or the first plot selected if multiple plots are being selected. However, you can show all cuts that are available by clicking on the **Show All** button. When the browser is showing all cuts, you can return to showing only the cuts on the selected plot by clicking on the button again. In either mode, you can select a plot by clicking in the browser.

Clicking on the **Add** button will apply the last selected cut to the selected plot(s). You can change the cut values by either

typing in a new value in the form or clicking and dragging on the slider. As you drag the slider, each plot dependent on the selected cut will re-display itself with the new cut value. This may include plots not currently selected. Also, a shaded region on the plot being used to display the cut variable will be updated. In the case of multiple cuts, the plots displaying the variable being used as a cut can also be cut with one of the other cuts. One adds a cut to a plot being used to display a cut in the same manner that one adds a cut to a normal plot.

The sliders in the cut inspector control the high and low range of the cut, or the the cut value when only one is needed. When the Zoom/Pan check box is selected, then the high slider narrows or broadens the cut range and the low slider moves the range up or down. Fine control of the sliders can be done by holding down the Alternate key while dragging. When this is done the normal response of the slider is scaled down by a large factor.

4.7 The Remaining main menu items.

The remaining main menu items are standard NeXTstep menu items and will not be described in this manual.

5. Programmer's Guide

This section describes the HippoDraw source code structure and how to incorporate HippoDraw in custom applications. The starting point for HippoDraw was the Draw application found in **/NextDeveloper/Examples/Draw** in the NeXTstep release 2.1 extended. An extract from the **README.wn** file in that directory is noteworthy…

> *This program is by no means an ideal implementation of a drawing program. It is intended to give example code for as many features available in NeXTstep as is possible in a single application.*

Thus, HippoDraw suffers from all the bugs, annoying features, and lack of features of the Draw application. In the development of HippoDraw, only the bare minimum modifications have been made to the Draw source code, thus with the next release of NeXTstep, one should be able to rapidly port HippoDraw.

HippoDraw is built from the application's NeXT Interface Builder file (.nib), some auxiliary files such as the TIFF files for the icons, an Interface Builder subproject and the Hippoplotamus package. Except for the main program written by the Interface Builder, all the sources are contained in this subproject directory (**Hippo.subproj**). This structure allows one to incorporate HippoDraw sources in custom applications. HippoDraw, the application, is just the default minimum application that uses them.

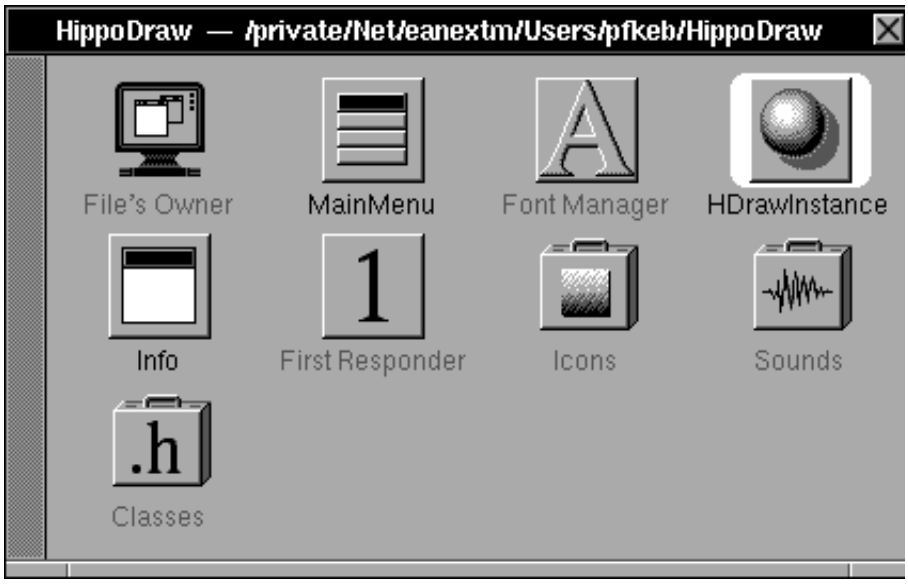
5.1 The application directory: HippoDrawSrc

The application directory contains the following files (in order they appear in the default **FileViewer**)…

- **help.hdraw** The help file from Draw.
- **hippoApp.tiff** The application icon.
- **hippoData.tiff** The document icon for .**hippo** files.
- **hippoDoc.tiff** The document icon for **.hdraw** files.
- **HippoDraw.iconheader** The iconheader file built by the Interface Builder**.**

- **HippoDraw.nib** The application's **.nib** file.
- **HippoDraw_main.m** The main program built by the Interface Builder.
- **Hippo.h** A header file built by the **Makefile** from the **Hippo.subproj** directory.
- **Hippo.subproj/** The subproject directory
- **IB.proj** The application project file.
- **Makefile** The **Makefile** built by the Interface Builder
- **Makefile.preamble** Extension to **Makefile** to include *hippo* header file for compilation and *hippo* library file for linking**.**
- **services.txt** The services file
- **.gdbinit** The **gdb** initialization file for developers.

The contents of the **HippoDraw.nib** file are shown in the graphic below. Besides the usual objects for all NeXTstep applications, there is an instance of the **HDraw** class named **HDrawInstance**. Developers must not change this name because serveral classes use the function **NXGetNamedObject()** to find this instance. This class can be found in the **Hippo.-subproj** directory. It is modified version of the **DrawApp** class of the Draw application. There are a number of menu items whose target is this instance of **HDraw.**



The **IB.proj** file contains very little. Besides the usually contents for all NeXTstep applications, it has **hippoNext** declared under “other libs” and **Hippo.subproj** under “Subprojects”. When building a custom application using HippoDraw sources, these two items are all that is needed.

5.2 HippoDraw Sources: Hippo.subproj

The Hippo.subproj directory contains source code of three types: unmodified Draw source, modified Draw source, and new Hippo source code. Each will be described briefly in the following sections.

5.2.1 Unmodified Draw Sources

The following files contain source code that should be identical to the original Draw sources…

- **Circle.***
- **Curve.***
- **DrawPageLayout.***
- **Graphic.***
- **GridView.***
- **Group.***
- **Image.***
- **Inspector.***
- **InspectorPanel.nib**
- **Line.***
- **PSGraphic.***
- **Polygon.***
- **Rectangle.***
- **Ruler.***
- **Scribble.***
- **SyncScrollView.***
- **TextGraphic.***

- **Tiff.***
- **cross.tiff**
- **draw.psw**

5.2.2 Modified Draw Sources

Modifications to the original Draw source code has been kept to a minimum. Usually, Draw classes were extended by sub-classing, but in some cases it was found necessary to modify the source. The following files contain modifications…

- **DrawDocument.*** This class has been modified slightly in order to handle the archiving of document files with *hippo* displays in them.
- **GraphicView.*** This class modified only slightly.
- **HDraw.*** In the Draw sources, these files were named **DrawApp.[hm]** and **Draw.nib**. The class defined in them was a subclass of Application. In order to incorporate HippoDraw in custom applications, **HDraw** was sub-classed from Object. Other modifications include adding the plot tools to the Tools panel.
- **IB.proj** This file was obviously modified to add the new source code.
- **Makefile** This file was obviously modified by the Interface Builder to include the new sources.

5.2.3 New HippoDraw Sources

This section lists the sources not found in the Draw sources. Some are useful classes outside of the HippoDraw context. Others are specific to HippoDraw. In the header files for each of these classes, each method (well, most of them anyway) is briefly described. With this release of HippoDraw, they are the only documentation available.

- **FineSlider.*** A subclass of the NeXTstep Appkit **Slider** class. This class extends the **Slider** functionality when the Alternate key is held down. Instead of returning a value corresponding the mouse position on the **Slider**, it returns a value as if the **Slider** had a reduced range.

- **HGraphicView.*** This class is a subclass of Draw's **GraphicView**. The added functionality is to support the HippoDraw's plots. The **.nib** file contains the Plot Preferences panel.
- **HTuple.*** This class maintains a *hippo* n-tuple.
- **Info.nib** The application's Info panel **.nib** file.
- **InspectAxes.*** This class manages the changing of a plot's axes and the **Axes Selection** inspector.
- **InspectBase.*** This class is a super-class of the HippoDraw inspector classes. Its purpose is to detect changes in the selected graphic and updated some instance variables.
- **InspectCut.*** This class manages the changes to cut parameters and the **Cut Options** inspector.
- **InspectData.*** This class manages the binding of n-tuple data columns to axes of the plots and the **Data Selection** inspector.
- **InspectPlot.*** This class manages the plot's graphic options and the **Graphic Option** inspector.
- **InspectStat.*** This classes manages the display of plot statistics and the **Plot Statistics** inspector.
- **InspectTuple.*** This classes manages the list of opened *hippo* n-tuple files (via the **HTuple** class) and the **Tuple List** inspector.
- **Legal.nib** This file contains a panel with some required legal information.
- **Makefile.preamble** This extension to the **Makefile** is to set the location of the required **hippo.h** file.
- **NewInspector.*** This classes manages the inclusion of Views from other inspectors and the selection of the View with a Popuplist.
- **Overlay.*** This is a sub-class of **Group** that manages Plots that have been overlayed.
- **PageMarker.*** This class is a sub-class of **Line** and draws the line indicating page boundary.

- **PGroup.*** This is an extension of the **Group** class done with the category mechanism for handling grouped plots.
- **Plot.*** This class is a subclass of **Graphic** and manages the drawing of a *hippo* display.
- **Plot1D.*** Obsolete class maintained for backward capability of reading older **.hdraw** files.
- **Plot2D.*** Obsolete class maintained for backward capability of reading older **.hdraw** files.

5.3 Building Custom Application with HippoDraw

To build a custom application incorporating HippoDraw sources, one needs to make an instance of the HDraw class in one's main **.nib** file, such as was done in the HippoDraw application. One will also need to connect menu items to this instance, and to **FirstResponder** using the **HippoDraw.nib** file as a guide. Perhaps the best way to proceed is to add custom application code to the HippoDraw application **.nib** file so that the starting point would be one with all the connections already made.

In most cases, the custom application will be creating a *hippo* n-tuple for which HippoDraw will responsible for displaying. The following source code fragment illustrates how the application can make its n-tuple known to HippoDraw and how to open the HippoDraw document window. Actually, two of the message return a useful object that is not shown in the example code. the message **[hdraw addTuple:nt]** returns an object of the class **HTuple** which will manage the n-tuple until it is closed. Look at the header files for this class for documentation. The message **[drawView addPlotOfTyple:5]** returns an object of type **Plot** which will manage the display. The parameter is one of the enumerated types defined in **drawtype_t** in the **hippo.h** file. You can message this object to change attributes of the display. Look at the documentation in the header file for more information. You only need to know about these returned objects if you want to do something fancier than the default displays.

```
nt = h_new(nt_dim);
hDraw = NXGetNamedObject("HDrawInstance", NXApp);
[hdraw addTuple:nt ];
if ( drawDocument == nil ) {
    drawDocument = [ DrawDocument new ];
}
drawView = [ drawDocument view ];
[ drawView addPlotOfType:5 ];
[[ drawView window ] orderFront:self ];
[ drawView display ];
```

5.4 Changes in the 1.1 Release

This section documents the changes in the 1.1 release as compared to the 1.0 release that could effect writers of applications that incorporate HippoDraw.

- The suppoted interface is described in the previous section. The former methods, **- setTuple:(ntuple) nt withNewDisplay:(int) dim** and sending a message to the **InspectTuple** instance are still in the current release, but may disappear in a future release.
- The menu item /Grapics/Align/Plots/Overlay now has the action **alignOverlay:**. The previous action, **alignSize:** has been kept for backward compatibility, but will be removed in a future release.
- The **Plot** menu item has been moved up one level and modified.