



User Defined Symbols: The S52 Raster Editor

By Bill Bergmann

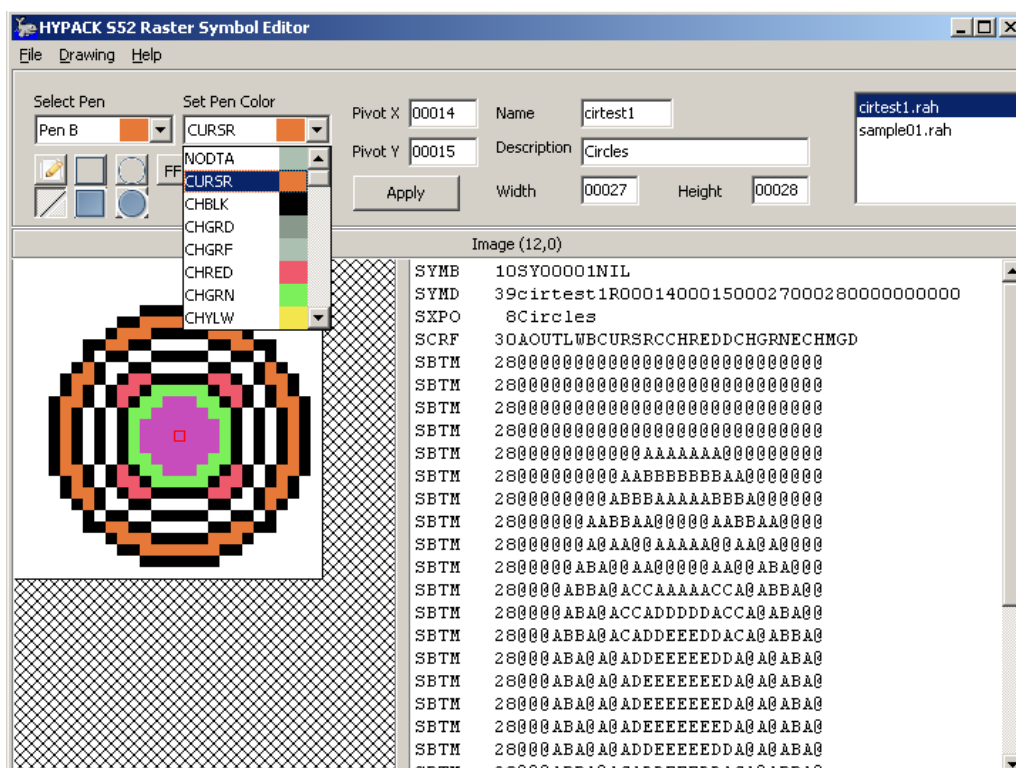
First we put the ability to create HYPLLOT Title Blocks in your hands. We have done it again, this time with chart symbols. There were many different factors we explored to provide this service. We ultimately chose compatibility with the S57 Presentation Library rules and symbols as developed under the IHO S52 standard as it is widely in use and fairly straightforward to understand, use and implement.

The S52 standard actually allows symbols to be defined in both vector and raster formats. The vector format is encoded in a form of HPGL commands. These scale nicely under different display resolutions, e.g. it will display equally well on your screen or a high density plot. They are, however, prone to Trial and Error cycles when trying to form valid commands to obtain a desired result. The raster format is analogous to a drawing in that it is composed of a rectangular grid of colored pixels. This makes it much easier to get a desired result, but it doesn't scale well when the display resolution is far away from the design resolution. Imagine zooming on a photograph. The tighter the zoom, the more block-like the image becomes. A common solution to overcome this shortcoming is to distribute multiple versions of a raster symbol e.g. low, medium and high resolution versions.

THE S52 SYMBOL EDITOR

Our S52 RASTER EDITOR supports the raster format only. It works roughly similar to a typical paint application.

FIGURE 1. S52 RASTER EDITOR



There is a canvas area where coloring operations take place. A number of tools are available to aid coloring your picture:

- The **pencil** allows coloring pixel by pixel.
- The **line tool** aids in drawing straight lines.
- There are also **rectangle and circle tools**, both filled and hollow versions, to quickly make these shapes.
- The **'FF'** tool, an acronym for Flood Fill, will fill all pixels with the active Pen color.

As you draw on the canvas, the resulting file which will be generated from the picture is constantly updated. This file view is next to the canvas. After a few lines of header information, you will see a series of SBTM encoded lines. These contain either an '@' sign, which is a placeholder for an unassigned color, or the letter of the Pen used when that pixel was assigned to it. The file format is very much like a paint-by-number scheme. When a Pen token is found, the color for the Pen is 'looked up' and the actual color is drawn at that location. This is analogous to a graphics image stored with a palette.

PENS

There are currently 10 Pens available, A through J, each of which can be assigned a color. The active Pen is the one shown in the Select Pen combo box. Whenever a drawing operation lights up pixels, the token of the active pen is inserted into the file at the proper location. Since it is a pen token which is stored at each location and not a specific color, it allows for easily changing the color of an existing drawing. For instance, if you have drawn a rectangle shape using Pen A, which is assigned the color CHRED, then change the color of Pen A to say CHGRN, the rectangle in your image will change to the new color.

COLORS

The available colors come from the S52 display standard. Each is named with a 5 character token, which with a little practice, make sense. (A few examples: NODTA - No Data, CURSR - Cursor, CHGRD - Chart, grey, dominant, CHGRF - Chart, grey, faint.) The 64 colors available, like the Pen tokens placed in the symbol file, don't refer to a specific RGB (Red Green Blue) value, but are instead selectors into a look up table. The active color scheme ultimately determines the actual color to be used. This is a powerful construct and was designed to alter displays based on lighting condition by simply changing the color table in use.

META INFORMATION

The following provides detail on those editor fields controlling information about the symbol, as opposed to the image itself.

- The **Width** and **Height** fields determine the extents of the image. By default, any part of the image not assigned to a specific Pen is considered transparent.
- The **Name** field contains the text which will show in image pick lists. This is to aid in selecting a proper image for a Target or DG2 element. You can see examples in the image below.

-
-
- The **Description** field allows storing arbitrary additional information about the symbol.
 - Use the **PivotX** and **PivotY** fields to orient the symbol when drawing. This point in the image will be aligned with the real world point of an object, for instance, a Target location, when being displayed.

Note: In keeping with the S52 standard, raster symbols don't rotate. If we allowed a rotation field, we would have to invent yet another storage format as the existing one hasn't a place to store rotation information.

SYMBOLS AVAILABLE IN THE TARGET AND DG2 EDITORS

Quick Access HYPACK® looks for user-defined symbols in your '<HYPACK INSTALL DIRECTORY>\bin\symbols' folder. All symbols in this folder will be displayed in a list box at the top right portion of the editor. Simply click a file name in the list and that symbol will be loaded into the editor, ready for modification. Don't worry, if an active drawing has been changed, you will be prompted to save those changes before the new one is loaded. If you happen to use the File->Open option to load a symbol from another directory, then all symbols in said directory will populate the list instead. Just be aware that the only symbols HYPACK® can 'see' are those in the default location.

SYMBOL USAGE

The S52 standard has the notion of a symbol being used for either display at a single location, or repeated to form a pattern in fill operations. Checking the '**Pattern**' check box makes available a few additional fields which are relevant only for Pattern Symbols. There are fields to specify **Fill Pattern**, **Spacing** and **Distance between Patterns**. You will notice the file tags also change between symbol and pattern mode, for instance SBTM tags becomes PBTM tags. You can now use your own patterns to fill area objects created in the DG2 editor.

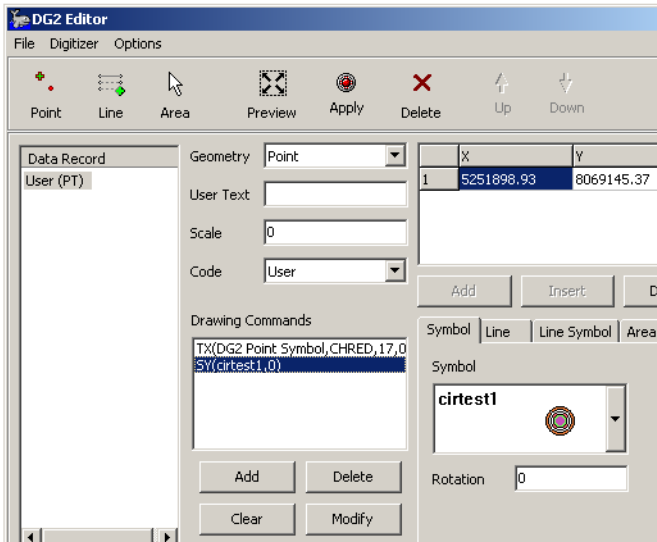
ODDS AND ENDS

There is a **Zoom In/Out button** to quickly switch between actual display size and a zoomed in size which is handy for editing. When changing any of the edit fields remember to press the Apply button in order to accept the changes into the editor.

The **location inside the image** is displayed above the canvas area as the mouse is moved. It is corrected for the current zoom level, and aids in setting a proper pivot point.

FIGURE 2. Custom Symbols in the DG2 EDITOR (left) and TARGET EDITOR (right)

DG2 Point Symbol



Circle Sample

