CPG MEMO NO. 10

March 31, 1983

Memo to: Bob Burns
From: Larry D'Addario
Subject: Graphics Software Support, From Electronics Div. Viewpoint

Copies to: Sandy Weinreb, Rich Lacasse

You and Sandy Weinreb have asked me to comment on the possibilities of purchasing or otherwise obtaining packages of software for supporting graphics terminals and other graphical output devices. This memo gives my thoughts on the needs of the Electronics Division in this regard, and also summarizes some materials I have reviewed on specific software packages which might be available.

1. General Requirements

The following are some of the needs of the Electronics Division, but I think that they are equally important to other users in the Observatory.

(a) Ease of use. There should be a subroutine library callable from a high level language which will have long term support within the Observatory. Since the only such language currently identifiable is FORTRAN, I shall assume that we need a FORTRAN-callable library. The set of subroutines should be sufficiently rich to allow finished plots to be generated with very little programming effort on the part of the user. This is pretty obvious, and most reasonable packages provide sufficient support. However, there exist some packages which are more business-oriented (emphasizing pie charts, bar graphs, and keeping track of the days in a month) than science-oriented, and these should be avoided.

(b) Portability. There are three such requirements: internal, import, and export. It is rather important to be able to share software among the Observatory's sites, so the same graphics support package should be installed on at least one computer at each site. (If this is not feasible, then at least the package should be accessible remotely from every site.) More difficult is the problem of portability outside the Observatory. We sometimes have a program which we would like to give to an outside colleague. If it requires graphics support, then either his computer must have the same library or a compatable library, or we must export the support library too. The latter is not feasible if the library is a commercial product subject to licensing agreements, and this has in the past caused the Observatory to write its own set of support routines for particular programs. This home-brew approach leads to subroutines which are hard to use from other programs; and making them more general seems to be an inefficient use of our programmers' time. Conversely, there is the importing problem: we'd like to borrow programs from our colleagues without having to create the necessary graphics support.

These requirements imply a need for standardization. If we can obtain an adequate package which is not subject to licensing restrictions (and there are such things--see below), then the internal and export requirements can be met, regardless of whether other people run that package. But in view of the import requirement, there is an advantage to choosing a package which is already widely distributed or which implements a widely recognized standard at the programmer level, so that different packages could nevertheless support the same application program.

2. Electronics Division Requirements

The computational needs of the Electronics Division are in my opinion well described in a memo by R. Lacasse dated February 8, 1982 (even though that memo explicitly addressed only Green Bank's requirements). All of the applications involving graphics can be met by monochromatic, line-drawing devices. I do not forsee a need for any raster-scan displays, either gray-scale or color; but this does not imply that we would not find a use for such things if they were available.

The software support required is that which would allow production of labeled plots with various kinds of lines, plotting symbols, grids, and scales. Methods of presenting three-dimensional data, such as contour plots and ruled surface plots, are also needed. For drafting and documentation, access to completely general line drawing routines is needed. At the level of the FORTRAN programmer, the subroutines should be device-independent; they should be linkable to device-dependent drivers, and these should be available to support all applicable devices at each site.

An important requirement which might be overlooked in an attempt to provide device-independence is this: we want to be able to write interactive applications programs, in which the user can view the results of many successive trial computations without leaving his program. This precludes an approach in which device-independent routines produce an image file in some "metacode", and this is later viewed by running a stand-alone, device-dependent translation program.

Some desirable, but (in my option) non-essential features of graphics software would be: (1) support for multicolor line drawing; and (2) support for the storage and retrieval of image files.

3. Standardization Trends in Graphics Software

From the literature, it seems that efforts toward standardizing graphics software have been underway since the early 1970's [1-3]. These have been directed toward both machine-independence and device-independence at the programmer level. A major milestone was the publication in 1979 of proposed specifications for a Core Graphics System by a working group of the ACM [4]. This has become known as the "1979 Core System" and a number of implementations exist. Further work has been done since then on various additions and enhancements, culminating in the current proposed standard known as "GKS" or Graphical Kernal System [5].

4. Comments On Particular Packages Now Available

(a) Precision Visuals, Inc. DI-3000. This is a family of software products, including some stand-alone programs, but mainly including a package of FORTRAN-callable subroutines which implement the 1979 Core System. It has been installed on a wide variety of systems, including various DEC, IBM, and HP computers. Device drivers are available for most machines that we use, including Calcomp plotters, Tektronics terminals, Versatec plotter, and HP plotters. I am told that this package has been purchased for use at the VLA. Its main disadvantage is high cost: August 1982 price for the lowest-level package was $8000 per CPU.

(b) NCAR-SCD Graphics System. The National Center for Atmospheric Research--Scientific Computing Division has developed a package of FORTRAN graphics subroutines which are device-independent and should run on a wide range of machines. This can probably be obtained by NRAO at little or no cost, and would probably meet the internal needs of the Electronics Division if it were to become the Observatory's standard. It might also satisfy the exporting requirement if no licensing agreement is involved. However, it does not appear to conform to any of the emerging standards, and so I would not recommend it for the long term unless cost becomes an overriding consideration.

(c) DOE: GWCORE. This is another implementation of the 1979 Core System, this time by the U.S. Department of Energy, available through the National Energy Software Center (9700 South Cass Ave, Argonne, IL 60439; 312/972-7250; #NESC-905/DB). I do not know any more details, but this seems worth looking into.

There are probably lots of other packages available that I haven't heard about.

REFERENCES

[1] Langhorst, F. E. and T.B. Clarkson III, "Realizing graphics standards for microcomputers." Byte, v.2, n.2, p.256 (Feb. 1983).

[2] Bono, P.R., J.L. Encarnacao, F.R.A.Hopgood and P.J.W. tenHagen, "GKS--the first graphics standard." IEEE Computer Graphics and Applications, July 1982, p.9.

[3] Warner, J.R., "Principles of device-independent computer graphics software." IEEE Computer Graphics and Applications, October 1981, p.1.

[4] "Status report of the graphics standards planning committee of ACM/SIGGRAPH." Computer Graphics, v.13, n.3, Fall 1979.

[5] "Graphical Kernal System (GKS)--Functional description." Draft International Standard ISO/DIS7942, version 7.02, August 9, 1982. American National Standards Institute (1430 Broadway, NY 10018; $28).