CPG MEMO NO. 23

National Radio Astronomy Observatory
Edgemont Road
Charlottesville, VA 22901 USA
(804) 296-0211, FTS=938-1271

1 November 1983

Memorandum

To: Computer Planning Group (CPG)

From: Don Wells Don Wells

Subj: Recent Graphics Standardization Discussions

About a month ago Bob Burns expressed a desire to investigate new graphics package options and asked me to find out what other national centers, especially KPNO and STScI, are doing for graphics packages. I called the relevant people and set up a conference call which was held on Wednesday 26 October. The participants were:

- KPNO: Doug Tody (designer of their IRAF system)
- STScI: Cliff Stoll (SDAS graphics specialist)
- NRAO: Burns, Greisen, Wells, D'Addario, Torson, Rots.

Bob explained our needs to Tody and Stoll and they described the decisions which their groups have made. In summary, both have adopted the National Center for Atmospheric Research (NCAR) plot package and both plan to implement the Graphical Kernel System (GKS) when it becomes available. While these decisions have been made for reasons which are internal to the respective projects, it is possible that there may be advantages for all if the national centers could coordinate to some extent. Probably a certain degree of portability of application code could result and our overlapping user communities would probably benefit from seeing some degree of consistency in plotting systems. Although I believe that these are important potential advantages and I hope that they can be realized, I think that we should make our decisions for our own reasons. As it happens, our reasons are almost identical to those of KPNO and STScI:

- we want to use national and international standards.
- we want a CPU- and device-independent package.
- the package should be in the public domain so our users don't have to pay licensing fees to use it.

In summary, it appears that the best course for us is the same one that KPNO and STScI have chosen:

- Obtain the NCAR package now and begin using it.
- Convert to GKS in the bottom layer of the package in about 2 years (NCAR will do most of the work).
- Evolve existing application code gradually toward GKS when it becomes available.

The NCAR Graphics Package

The NCAR package is a fairly comprehensive collection of machine-independent (Fortran-66) subroutines for scientific plotting. I regard it as a satisfactory, although unexciting, solution to the conventional graphics problems of scientific data analysis, especially of the Calcomp plotter variety. There is only limited support for 3-D data structures. The standard release is batch-oriented. It writes a "metacode" (device-independent) plot file which is then translated for particular devices by device-dependent code.

NCAR supports their plot package for their own internal uses and distributes it to other institutions on an "as is" basis for a $200 charge. The package comes with a substantial manual. NCAR does not care what other institutions do with the package. For example, apparently they don't care if it is re-distributed by recipents. This is important for us because we don't want NRAO's users to have to pay any licensing fees in order to use our application software on their own computers. KPNO and STScI have the same motivation. The NCAR package is already widely used in the university community. KPNO has used it on their CDC computer since about 1976.

I called the people at NCAR to inquire about the present situation and future prospects for their package. The person I spoke to was John Humbrendt (303-497-1286). [The graphics specialist at NCAR is Lofton Henderson, but he was not available.] Humbrendt told me that the people at NCAR want to move toward GKS standardization. They currently have two people assigned to the GKS conversion project and expect to have preliminary releases of a GKS version of their package next year. This GKS version will be in Fortran-77 and will conform to "level 0A" of the draft GKS standard. This means that it will still be batch-oriented. They expect to move toward a full interactive GKS implementation eventually. A major motivation for KPNO, STScI, and us to choose NCAR is that they will provide a migration path to GKS.

Although the "standard" machine-independent NCAR package does not support interactive graphics, we learned during the conference call that NCAR distributes a version which is adapted for Vax/VMS which does support cursor readback for graphics terminals. Clearly we should order this version for our VMS machines in addition to the standard distribution.

GKS - The Graphical Kernel System

No discussion of graphics packages these days can be complete without mention of the national and international standardization situation. This subject often seems analogous to the medieval pursuit of the Holy Grail. Committees have been meeting for years to try to reach agreement. The ANSI committee produced a draft in about 1977 which came to be called the "Core" standard. It was only a <u>draft</u> standard, but commercial vendors implemented it anyway. [Jim Torson purchased the Precision Visuals implementation of Core for the VLA recently.] But the ISO committee kept working and produced the GKS draft, which is in some sense a further extension of the ideas of Core. The initial

implementations of GKS were done in Germany and recent work is going on at the Rutherford-Appleton Lab in the UK. Apparently the RAL work is closely associated with the Starlink project at RAL. (Starlink is the network of Vaxes installed at almost all UK astronomy centers.) Pat Wallace (Starlink Project Manager) tells me that he has arranged a deal in which he is allowed to distribute GKS code to all astronomy groups without license. All of this would not be very interesting in the USA except for one big fact:

The ANSI committee has dropped Core in favor of GKS!

The way is finally open for an international graphics standards agreement. The commercial vendors are beginning to jump on this bandwagon: Tektronix recently announced that they will market and support GKS for use with their Plot-10 graphics library. They intend to continue to support code based on Core but expect to see a gradual transition to GKS. GKS is not perfect: it does not at present support 3-D, whereas Core does. On the other hand, GKS does support raster graphics devices rather well. Apparently everyone is assuming that GKS will evolve to include anything which is now missing from it. Also, it will be important to know how easy it will be to convert existing applications based on Core to use GKS.

It is natural that many people would ask why this subject (GKS) is thought to be so important. The answer is that our scientific applications are heavily dependent on graphics support. The portability of our code depends on having portable graphics support. New graphics devices continually appear and so we want our code to be device-independent. But even if we can utilize new devices in principle, in practice we repeatedly face the problem of providing the necessary device support code for the new devices. What we would really like is for the device vendors to do the work! Also, why should we have to install the graphics package in the environment of each new operating system which we encounter? We want the OS vendors to do that work! The principal point of GKS is that it provides a precisely defined interface standard which all of the vendors can implement and which will assure the compatibility of their products. The existance of the standard will also allow vendors to market devices which accept the GKS metacode format directly, thus taking computing load off of host computers and giving higher performance. Another kind of benefit is that GKS recognizes the profound importance of raster-style graphics for the future and offers strong support for it. For example, it appears that GKS may be able to integrate support for digital imagery (rasters) into its support for pure graphics. This would make image display software as portable and device independent as graphics software. It is the expectation of these kinds of future benefits which causes all of us to want to move toward GKS as a long range strategy. But during the next few years we must adopt temporary measures while we wait for GKS to become a practical option. The trick is to choose temporary schemes in such a way that they will offer a graceful transition to GKS in the future. It appears that use of the NCAR plot package could be one component of our transition strategy.

The AIPS Graphics Situation

AIPS is the largest body of application code at NRAO which is likely to live into the GKS era and so it is natural to ask what will be the AIPS response to the advent of GKS. The first thing to realize is that at present we have no immediate incentive to make a conversion to GKS or NCAR or anything else in AIPS ("if it works, don't fix it"). AIPS already utilizes most of the design concepts of portable device-independent graphics packages (e.g., metacode files). It is conceivable that AIPS might have profited by having chosen to utilize some portion of the NCAR design when the design decisons were made some years ago, but I can see no reason to utilize NCAR code in AIPS at this time. The sole exception to this conclusion is that we might want to examine the NCAR contouring algorithms to see if they might offer some advantage over the existing AIPS algorithms.

I assume that someone will eventually want to install AIPS on a supermicro workstation with GKS support embedded in the OS, or else they will want to use a graphics terminal or image display which accepts GKS metacode directly. At that point the obvious approach will be to devise an interface layer underneath AIPS to adapt to GKS for such systems. I can see no reason to do it in the near future. I do think that we must continue to watch the GKS situation. I expect that at some point in the future a continuing estimate of cost versus expected benefit will show that it is advantageous to adopt GKS for use inside AIPS. My own guess is that this will be 2-5 years from now, assuming that GKS development continues as it has in the recent past.

Followup Meeting

During the one hour conference call meeting on 26 October several participants expressed a desire to convene a <u>small</u> followup meeting at which the national center plans and implementations could be discussed in more detail, without the restrictions of the conference call format. Bob Burns will arrange the meeting, which we expect will be held in January. We hope that a representative of NCAR can attend in order to talk to us in more detail about their plans.

enc: Summary of NCAR routines, two pages of sample plots.

Graphic Display

(GRAPHICS) This package consists of over 50 files, including utilities for graphs, labels, contour plots, surface plots, 3-D projections, streamline plots, vector field plots, and maps. The utilities call entries in a plot package that produces device-independent metacode, which may be translated into actual plotting instructions for any graphics device. The system is implemented entirely in FORTRAN. Detailed information and instructions sufficient to guide implementation of the system is found in the manual, "NCAR Graphics Software", which is included with any request for the graphics package.

The basic system plot package that supports the system of utilities will be found in a file named PLOTmn, where m and n here stand for small integers that designate a configuration for a computer with m characters per default type INTEGER word and n bits per character. Besides TESTLIB files of portable testing routines, the other files that constitute the NCAR graphics package are

AUTOGRPH	DASHSMTH	PWRITXC1	SCROLL
CONCOM	DASHSUPR	PWRITXC2	SRFACE
CONRAN	ENCD	PWRITXD1	STRMLN
CONRAQ	ERPORT	PWRITXD2	SUPMAP
CONRAS	HAFTON	PWRITXNT	SUPMAPDT
CONREC	I1MACH	PWRY	TEST12
CONRCQCK	ISOSRF	PWRZI	TESTPLOT
CONRCSPR	ISOSRFHR	PWRZS	TESTSPP
CONTERP	MCTRPORT	PWRZT	THREED
DASHCHAR	MCTRPRNP	Q8QST4	VELVCT
DASHLINE	PWRITX	R1MACH	WINDOW

(VAX GRAPHICS) A version of the NCAR graphics package tailored for DEC VAX computers and the VMS operating system is available on a separate tape (for a separate $200 charge). This version will be easier to implement on VAX computers than the standard portable graphics package. The tape will be sent in BACKUP format with an accompanying description and VAX implementation guide.

AUTOGRPH Draws and annotates curves or families of curves. The standard version of this package requires DASHCHAR. The smoothed version requires DASHSMTH.

CONCOM A set of routines used by the CONRAN and CONRAS packages.

CONRAN Contours irregularly spaced data, labelling the contour lines. The standard version of this package requires DASHCHAR. The smoothed version consists of this file with DASHSMTH.

CONRAS Like CONRAN, but bigger and slower because lines are smoothed and crowded lines are removed.

CONRAQ Like CONRAN, but smaller and faster because it has no labelling capacity.

CONREC Contours two-dimensional arrays, labelling the contour lines. The standard version of this package requires DASHCHAR. The smoothed version, CONRECSMTH, consists of this file with DASHSMTH.

CONRCQCK Like CONREC standard, but faster and smaller because contours are unlabelled. This package shares entry names with CONREC, so they cannot both be included in a binary library.

CONRCSPR Like CONREC, but bigger and slower because contours are smoothed, labelled, and crowded lines are removed. This package shares entry names with CONREC, so they cannot both be included in a binary library.

CONTERP A set of routines used by the CONRAN, CONRAQ, and CONRAS packages.

DASHCHAR Software dashed line package with labelling capability.

DASHLINE Like DASHCHAR, but smaller and faster because it has no labelling capacity. This package shares entry names with DASHCHAR, so they cannot both be included in a binary library.

DASHSMTH Like DASHCHAR, but bigger and slower because lines are smoothed. This package shares entry names with DASHCHAR, so they cannot both be included in a binary library.

DASHSUPR Like DASHCHAR, but bigger and slower because lines are smoothed and crowded lines are removed. This package shares entry names with DASHCHAR, so they cannot both be included in a binary library.

ENCD A subroutine for label encoding, referenced by all CONREC... packages.

ERPORT A portable error handling package, used by CONRAN, CONRAQ, and CONRAS. This is a nonproprietary part of the PORT Mathematical Subroutine Library from Bell Laboratories.

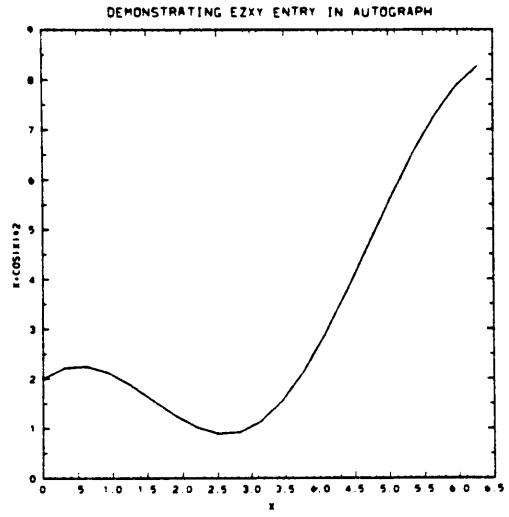
HAFTON Halftone (gray scale) pictures from a two-dimensional array.

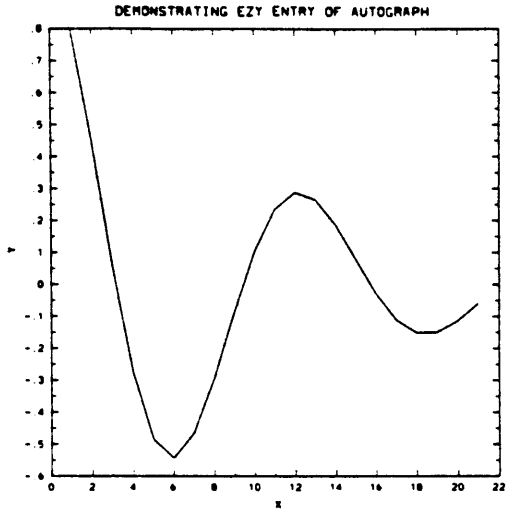
I1MACH Master version of the required support routine for returning integer-valued information on the host machine environment. All known versions exist as comment cards.

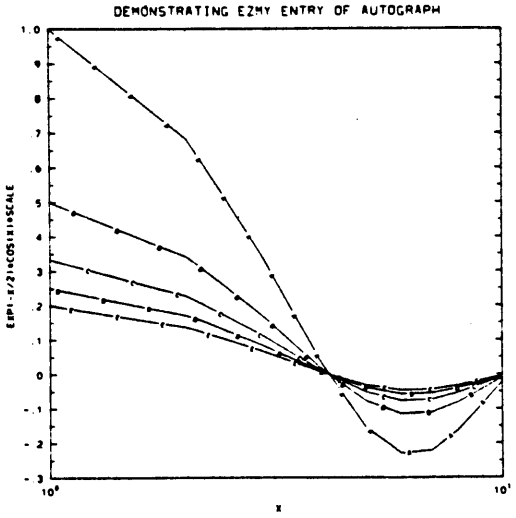
	This is an NCAR version of a nonproprietary part of the PORT Mathematical Subroutine Library from Bell Laboratories.
ISOSRF	Iso-valued surfaces (with hidden lines removed) from a three-dimensional array.
ISOSRFHR	Iso-valued surfaces (with hidden lines removed) from a high resolution three-dimensional array. Note: this package is relatively unportable for other than Control Data 60-bit/word computers.
MCTRPORT	A portable metacode translator "shell": the user sets a handful of constants relating to the host processor and target graphics device, provides 3 specified functions for boolean manipulation, and provides a minimal interface to intended graphics device at a single location.
MCTRPRNP	A portable translator that drives a line printer as a crude graphics device.
PLOTmn	A version of the NCAR system plot package that is configured for a computer with m characters per default type INTEGER word and n bits per character.
PWRITX	Draws fancy characters, using the Hershey database.
PWRITXC1	Part I of card image representation of PWRITX Complex font digitization.
PWRITXC2	Part II of Complex font.
PWRITXD1	Part I of card image representation of PWRITX Duplex font digitization.
PWRITXD2	Part II of Duplex font.
PWRITXNT	Portable program for turning card image fonts into binary PWRITX database.
PWRY	Simplest software characters.
PWRZI	Draws characters in three-space, for use with ISOSRF.
PWRZS	Draws characters in three-space, for use with SRFACE.
PWRZT	Draws characters in three-space, for use with THREED.
Q8QST4	A dummy copy of an accounting subroutine that is called by NCAR utilities.
R1MACH	As for I1MACH, but pertains to real-valued constants. This is an NCAR version of a nonproprietary part of the

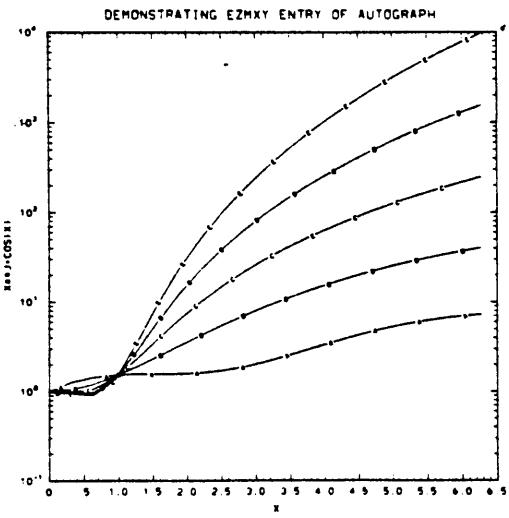
	PORT Mathematical Subroutine Library from Bell Laboratories.
SCROLL	Title-producing package used in making movies.
SRFACE	Three-dimensional display of a surface (with hidden lines removed) from a two-dimensional array.
STRMLN	Plots a representation of a vector field flow of any field for which planar vector components are given on a regular rectangular lattice, displaying both field direction (via lines of flow containing arrowheads and feathers) and field magnitude (based on distance between those flow lines).
SUPMAP	Plots continental and/or United States state outlines according to one of nine projections.
SUPMAPDT	Data base, containing continental outlines, etc., for SUPMAP.
TEST12	Tests the implementation of most of the "required locally-implemented routines", which the user must supply to support the otherwise portable NCAR graphic system (there are now actually 14, contrary to the implied 12).
TESTPLOT	A small program to test plot package implementation by generating metacode for two simple frames. Octal and hexadecimal dumps of the proper resulting metacode are included as comment cards.
TESTSPP	A program to exercise all entries in the system plot package.
THREED	Provides three-space line drawing capabilities, with entry points equivalent to the line drawing entry points of the system plot package.
VELVCT	Two-dimensional velocity field displayed by drawing arrows from the data locations.
WINDOW	Provides a clipping capability for lines extending outside a user-defined window, thus allowing part of a picture to be plotted without distortion or overwriting near the edge of the picture.

SAMPLE PLOTS: AUTOGRAPH



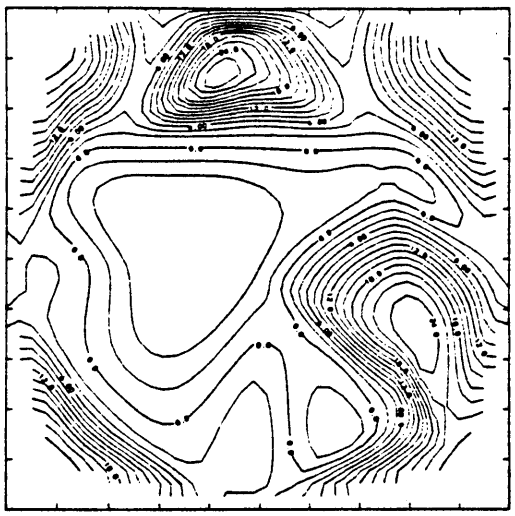






SAMPLE PLOTS: CONRAN, CONRAQ, CONRAS

DEMONSTRATION PLOT FOR CONRAN



DEMONSTRATION PLOT FOR CONRAQ

DEMONSTRATION PLOT FOR CONRAS