Eurographics '83 focuses on GKS implementations

Ware Myers, Contributing Editor

The significant development at Eurographics '83, according to Program Chairman Paul J. W. ten Hagen of The Netherlands, was the positive interest in the Graphical Kernel System, now a Draft International Standard.

The conference, held at the University of Zagreb, Zagreb, Yugoslavia, August 31 through September 2, attracted 265 delegates from 26 nations—about 65 more than anticipated. The conference organizers were especially pleased by the presence of 92 Yugoslavian participants, roughly three times as many as from the runner-up countries, West Germany and the United Kingdom, and they felt that the conference succeeded in one of its principal aims—to bring computer graphics to Yugoslavia.

GKS is positively significant, ten Hagen explained, because of the large number of people working on it, the many implementations now in progress, and the related work on data bases, communications, and hardware. In a separate interview, F. R. A. Hopgood of Great Britain, an invited lecturer and coauthor of a book on GKS, made the same points. Six of the 33 papers presented at the conference dealt with some aspect of GKS.

The reason for the interest in GKS, in ten Hagen's opinion, is that the standard gives the computer graphics community "the maximum possible independence from hardware." A modest computer graphics industry, based on microprocessor technology, is emerging in Europe, he says. It is building intelligent graphics workstations, many of which are built around GKS.

Raster graphics. On the other hand, the absence of papers on raster graphics drew attention to the fact, as ten Hagen says, "that the rapid and spectacular development of the raster graphics area is not well represented in Europe." At this time it is "purely a North American affair."

Europe uses some graphics equipment but does little research and development in the area. Spectacular applications such as movie animation are moving rapidly in the US, but not in Europe. A similar difference exists in the area of simulation. Perhaps next year's conference, to be held in Copenhagen, September 12-14, will reveal more work in this area, ten Hagen speculates.

At any rate, the existence of Eurographics, now four years old, has led to "rapidly growing cooperation in terms of working groups getting together and sharing results," ten Hagen noted. "That will have further effects on what will come out of graphics in Europe. GKS is an example of a European effort that has been accepted in the United States."

Europeans are trying to do something similar in other areas, such as communications and graphics algorithms. They will try very hard, ten Hagen believes, to compete at putting graphics workstations into networks. Unfortunately, raster technology is not catching on very fast. Moreover, Europeans don't have databases where they should.

Still, these key issues "will lead into the integration problem—CAD and CAM," ten Hagen concluded. Integration must be supported by a sophisticated database that can contain pictures of geometrical information. These problems are beginning to receive some attention.

Graphical Kernel System. The proposed standard defines a set of functions that serves as the interface between one or more application programs and a configuration of input and output devices. This set of functions is the kernel, and it is independent of the specific characteristics of both the application programs and the devices. The functional characteristics of the devices, such as graphics workstations, are held in a device description table. When a function appears, it is checked against the table. If the device has the function, the pertinent information is passed directly to it. If it lacks the function, the information is mapped to the best available function.

GKS implementations. Gunther Schmitgen describes an implementation called GKS-300 based on GKS version 7.0, level 2c, for the Siemens graphics workstation Sigris. Sigris is a Siemens Series 300 process computer with graphics peripherals, including a storage tube and a graphics tablet. Schmitgen is also planning to interface GKS-300 to a raster display. The kernel is written in Pascal with some functions...
in assembler. On the application-program side, GKS-300 interfaces to programs in Fortran and Pascal.

Sergio Antoy and Giuliana Dettori of Italy set forth some of the problems in binding GKS to Pascal. Making GKS functions available to programs written in a specific language is referred to as language binding. Antoy and Dettori described problems related to the use of a library, to the passing of array parameters to procedures, and to the definition of suitable types for GKS data. They concluded that their "minimal strategy...seems the best one because it offers simplicity and reliability."

At the Computer and Automation Institute of the Hungarian Academy of Sciences, a small group* had nearly completed XGKS, a multitask implementation of GKS, at the time of Eurographics '83. The device-independent part occupies about 40K bytes. The interface to the C language is about 2K bytes, and the interface to the metafile output workstation is about 10K bytes. Work is underway on device drivers for the Tektronix 4010 and the GD80, a Hungarian intelligent workstation. The group plans an interface to Pascal.

When Exploration Consultants Limited started development in 1982 of a graphics facility for its petroleum industry consultancy, GKS was just emerging as an ISO Draft Proposal, Howard K. Watkins of the United Kingdom reported. The consultancy wanted a graphics package that could interface to its existing mainframe computers and peripherals and also act as a user interface to GKS. These aims have been achieved.

*I. Herman, T. Toluiny-Knefely, and A. Vincze.

The package, which is named Granite, is available on the Data General MV8000 minicomputer and is being installed on a Hewlett-Packard 9000 32-bit desktop machine. Output can be directed to the Tektronix 4014, Visual 500, Calcomp 748 flat-bed plotter, Calcomp 1051 drum plotter, or Versatec 8236 electrostatic plotter. As a user interface to GKS, Granite has reduced the number of concepts the user must understand, at the same time providing a half-dozen extra functions, such as page layout.

There are many types of displays, and GKS must ultimately interface to each one through a device driver. Drawing on his experience with five different raster graphics systems, Joachim Rix of West Germany has sorted out the functions that could be included in a driver architecture common to all display devices. He believes it is possible to define a set of modules that will incorporate these kernel functions, which include segment transformation, workstation transformation, and attribute bundles. In addition to these functions, he is sorting out the functions specific to each class of device—in this case the raster device. These functions include scan conversion, color look-up table, pattern table, etc. The kernel and device type-specific functions together define the minimal driver.

"A database has to provide (1) a concise and understandable representation of the state of a system at any time, (2) efficient and high-level access to information, and (3) simple and consistent update as the system changes state." M. Rudalics of Austria pointed out. His intelligent graphics terminal's intermediate database combines two concepts on one storage level: (1) a workstation-dependent segment storage, as described in GKS, and (2) a display list for a workstation providing output of graphic primitives in store-and-refresh mode.

References


Invited lecturer F. R. A. Hopgood of the United Kingdom believes single-user workstations herald a major change in computer usage—a change that can result in much improved user interaction and productivity.