CERN-Data Handling Division
DD/82/11
J. Bettels
R. Brun
July 1982

COMPUTER GRAPHICS IN HIGH ENERGY PHYSICS

(Submitted to IEEE Computer Graphics and Applications)

Computer graphics is an invaluable aid to the design and interpretation of subatomic particle experiments conducted at the CERN laboratory on the Franco-Swiss border.

Computer Graphics in High Energy Physics

Jürgen Bettels and René Brun

European Organization for Nuclear Research

High energy physics is the study of the subnuclear constituents of matter, constituents that, for historical reasons, are still often called "elementary particles."

They are extremely small, about 10^{-13} centimeters in diameter, and mostly very short-lived, some lasting only 10^{-40} seconds. To observe such particles and their interactions, physicists need machines that let them look, with very high resolution, deeply into matter. These machines probe matter with beams of accelerated protons or electrons of such high energy that their wavelengths are short enough to achieve the desired resolution. Particle beams are produced in very large accelerators located in the US,

Western Europe, the USSR, and Japan. Building such a machine typically costs several hundred million dollars; a staff of at least 1000 people is needed to run it.

Largely because of these costs, it is important that nations and institutions collaborate to build the machines and perform experiments on them. The European Organization for Nuclear Research (CERN), for example, is a collaboration of 12 European countries maintaining a large accelerator laboratory on the Franco-Swiss border near Geneva (Figure 1). The main facility of CERN is an accelerator, called the Super Proton Synchrotron, or SPS, which is located in a deep underground ring tunnel

Figure 1. Aerial view of the CERN site in the Geneva Valley indicating the locations of the SPS ring tunnel and the future LEP ring tunnels.

seven kilometers in circumference, where physicists from Europe and from other areas of the world are carrying out experiments. A typical experiment can involve six to ten teams from different universities, perhaps totalling 30 to 60 physicists.

Last year the construction of a new accelerator, called LEP, was started at CERN. It will begin operating by 1988 and will be located in a deep ring tunnel 27 kilometers in circumference, stretching from the Geneva airport to the slopes of the Jura mountains in France. The experiments will be carried out in underground halls (Figure 2).

Data processing is important in particle physics experiments because the volume of data is very large and the computing requirements are complex. It is quite normal for a single experiment to record enough data to fill several thousand tapes per year. As a consequence, every high energy physics laboratory has a large data handling center, even though more than half of the data is processed at facilities of the collaborating universities or research institutes. Furthermore, the experimental detectors require complex control and monitoring by digital logic and computers. Recent experiments have used several 32-bit and 16-bit minicomputers and over a hundred 16-bit and 8-bit microcomputers.

The similarities in experimental devices and problems, and the close international collaboration, favor the adoption of common software methods, which often develop

into widely used packages. The European Committee for Future Accelerators (ECFA) has set up a study group to develop standard software tools for some of the more common functions, in anticipation of preparing experiments at the future LEP accelerator. A common graphics tool is one of the subjects being pursued by this group.

In a large research laboratory there are naturally many important applications of computer graphics, such as accelerator control and CAD, which will become an important tool in many aspects of the civil, mechanical, and electrical engineering in the LEP design and construction of the new LEP accelerator. In this article, however, we restrict the discussion to the application of computer graphics to experiments in physics alone.

Data flow in a high energy physics experiment

The collision of particles with each other or with a target produces a varying number of secondary particles that leave the interaction point in all directions. The total of the tracks made by these particles is called an "event." The data analysis of an experiment consists in calculating the topology of each individual event to determine physical quantities such as momentum and mass, and in deriving more complex physical measurements statistically from a large number of events.

Figure 2. Artist's impression of an underground experiment hall as it will be built for the LEP accelerator.

Figure 4. UA2 experiment, detector layout, and Monte Carlo event of a proton colliding with an antiproton, as seen on the screen of a Merlin workstation.

detector and how many background events will produce the same signal (Figure 5). Monte Carlo calculations will establish the level of the expected background. When the background is subtracted, the observed signal becomes much clearer. It is evident that these calculations often require large amounts of computation time, and therefore the programs are normally run in batch mode, producing metafiles for graphical output. We inspect these metafiles by displaying the generated pictures, mostly on DVST screens, or simply by plotting them. This may result in modifications to the detector, the generation of new Monte Carlo events, or the display of results. Given enough computing power, a session in front of an interactive graphics terminal in closed loop with the programs could speed up this process considerably. But the computing requirements are so taxing that for the present it is possible to treat only parts of the process in this way.

Once the detector setup has been decided, it is time to develop the event reconstruction programs. This is largely

a pattern recognition problem, but it is complicated by several factors. A particle generated in a collision typically travels through and leaves signals in several parts of the detector, such as wire chambers, scintillators, and Čerenkov counters. All the signals must be combined to reconstruct the path of the particle. At the same time, signals from other sources, crossing tracks, and tracks of cosmic particles must be filtered out. Furthermore, if the detector, or parts of it, is in a magnetic field to allow momentum measurements, the charged particles will leave curved tracks. Particles may also collide again with matter in the detector, be deflected, or be absorbed in a reaction that produces even more particles.

The reconstruction program must take care of all these effects. As conditions become very complicated, visual control is the most effective way to develop and debug the program. Usually in the preparatory phase of the experiment, a large amount of Monte Carlo data is generated, simulating the signals in the various elements of the detector, including the disturbing effects of background, false signals, missing signals, and errors in signals. The simulated data passes through the reconstruction program, and the reconstructed particle tracks, together with the simulated signals, are displayed in various projections, by means of several different graphics packages.

CERN uses its own package, GD3, which normally produces pictures in batch runs and enables the user to later examine an intermediate metafile frame by frame on a DVST screen or a pen plotter. The Rutherford Appleton Laboratory in England most commonly uses the Smog graphics package for this type of work (see page 68 for more details). At DESY, the projected views of signals and tracks can be studied interactively on IPS display stations (see page 66) using Tektronix 611 and 619 monitors and TSO on the IBM mainframes.

In many cases the human eye is very quick to recognize a faulty assumption made by the pattern recognition program. The researcher then tries to improve the techniques of filtering and track matching in order to achieve the highest possible "pass rate," an important measure of the quality of an experiment.

The limitations of this technique are clear when the detector becomes too complex and the track multiplicity too important, as in some recent experiments. Although the human eye is extremely good at recognizing the correct fitting of a track to a number of signals in one projection, it is very poor at making the same correlation in a different projection. To overcome this problem, we tried a different technique for proton/antiproton collider experiments at CERN. We developed the Merlin facility,¹ which consists of Megatek 7210 calligraphic terminals connected to a dedicated VAX/11-780 computer. On the Merlin graphics stations, the picture of the detector, the signals, and the reconstructed tracks, all defined in 3-D, can be rotated and zoomed smoothly in real time, so that we immediately get a very good impression of the relative location in space of tracks and signals. The displays are controlled by the reconstruction program run interactively on the host VAX. If a simulated event fails, we can adapt the program parameters and try a new, maybe more successful reconstruction.

Figure 5. A Monte Carlo generated histogram with superimposed background.

Applying 3-D graphics with real-time transformation capabilities to simulated data has proved very useful in developing and debugging the program chain before the actual experiment begins. This method will be often used to prepare experiments for the LEP accelerator.

The data-taking stage

Running an experiment requires supervising the functioning of a very complex particle detector (Figure 6) with many thousands of individual components. Clearly, not every physicist in the experiment can be familiar with every detail of the apparatus, so some generally understandable monitoring tools have to be provided.

Figure 6. A typical detector set-up for a fixed target experiment (NA4 experiment at CERN).

Figure 7. The monitoring console in the control room of a high energy physics experiment.

One of the on-line minicomputers usually runs a monitoring task (see Figure 7). At regular intervals, this program takes some subset of the incoming data and writes it to a file, which is regularly inspected by another program, often called a presenter, that shows in a graphical form those quantities relevant to the good functioning of the detector. The supervising physicist can command the display of statistical information about almost any quantity worth monitoring, such as voltages, gas flow, wire hits, and many others. In general the information is displayed in the form of histograms, but some experimenters prefer to display results of already reconstructed data showing the tracks of the events.

Most of today's experiments at CERN use on-line 16-bit minicomputers, with specialized graphics packages, often written in machine language to compensate for the limited memory of these computers. At CERN we are trying to provide graphics tools that are more compatible with those used in the later data analysis stage. Our current three basic modules are the histogramming package Hmini, which is compatible with the programs Hbook and Hplot (for more software details, see page 68) and can set up any histogram or data plot; the presenter program that transforms this data into pictures; and the graphics package MGD3. This graphics package is a subset of the GD3 functions written for the minicomputers on-line to experiments. It is considered to be a provisional solution and will probably be replaced when the proposed ISO GKS (Graphical Kernel System) standard can be implemented.

Event reconstruction

Once the first beam is brought into the detector and the first data is written to tape, the busiest months are ahead. The reconstruction program is tried out on real data for the first time, and minor or major problems, depending on how well the program was prepared, will show up. When the reconstruction program has been sufficiently refined, the recorded data, event by event, is passed through the program in quantity and the results are written to the data summary tape.

Because the problems of pattern recognition can never be solved totally, the program will reject a number of events, mostly because one or several tracks could not be filtered out of the background. Many experiments will try to rescue such events by extracting them from the original data tapes, displaying or plotting them, and, through careful studying of the faulty tracks, manually "helping" the pattern recognition to a correct result. Because of the time involved, this can, of course, be done only to very few events or to those of special importance.

"Scanning" is another procedure that requires looking at the graphical presentation of the reconstructed events. At CERN, GD3 metafiles containing images of event projections (Figure 8) are generated and then scanned in a semi-interactive way; that is, they can be scaled and translated. The IPS graphics system at DESY is much more interactive, allowing different projections or presentations in different coordinate systems to be requested, a capability particularly useful for clearly identifying an event.

In general, to be useful, such pictures must be of relatively high resolution; the ability to distinguish about 1000 points across the screen is probably the minimal requirement. The number of vectors and points needed to show the curved particle tracks and the signals is in the range of a few thousand to ten thousand. The DVST device has been the most economical way to display such pictures, and this is largely why other types of displays have seldom been used until now. Until very recently, raster displays were of too low resolution and their staircase effect is unsuitable for distinguishing small angle deflections of curved tracks.

As mentioned earlier, the Merlin facility has already proved useful in the preparation phase of the proton/antiproton experiments and even more useful in the event reconstruction phase, which started at the end of 1981. Because of the multiplicity of the particles produced in the collision, most proton/antiproton events have very complicated topologies (Figure 9). To check the results of the reconstruction and kinematics programs for verification or scanning purposes, it is important to see the relative position of the particle tracks in three-dimensional space. The 3-D real-time transformation capabilities of the Merlin display stations proved very effective for this purpose.

We are currently developing software at CERN that allows a much more flexible strategy for viewing the data, using a highly structured data base to represent the graphical information. A number of institutions in Europe and the US, collaborating in the proton/anti-proton experiments, are now setting up similar facilities. We expect them to be indispensable for experiments at the future LEP accelerator.

Figure 8. Event projections in the NA2 spectrometer at CERN.

Figure 9. The proton beam colliding with an antiproton beam, as registered in the UA1 detector at CERN. The picture, taken from the Merlin screen, shows a 3-D representation of the detector and the signals.

Graphics in the data analysis stage

All along the data processing chain, the programming language most commonly used is Fortran, and the output information usually consists of statistical distributions.

A typical production job analyzing several thousand events requires about one hour or more of CPU time on the IBM 3081 or CDC 7600. This produces between 50 and 100 statistical distributions in the form of one- or two-dimensional plots.

The most commonly used histogramming package is Hbook.²

Hbook and its output options. Hbook is a package consisting of a few hundred Fortran subroutines that can symbolically define, fill, and output on a line printer one- and two-dimensional density estimators in the form of histograms, scatter plots, and tables. About 120 of the subroutines are directly accessible from the program via Fortran calls. Some interesting features of Hbook are

- The basic operations are a few subroutine calls that can be learned in half an hour from three pages of documentation.
- The ability of Hbook to manage its own virtual memory on mass storage in a fully automatic way, hidden from the user, can save space. The archiving option separates different phases of the histogramming process into job steps.
- Histograms and plots are represented on the line printer in a standard format that contains the picture and some numerical information. Several options can modify the presentation mainly for one-dimensional histograms. The user can also add or suppress information, choose a different graphic presentation, and modify the mapping of histograms on output pages.

When the line printer's resolution is inadequate, the physicist has several other graphics packages at his disposal. In general his choice depends on the institution where he is working and the facilities available.

Hplot³ is a package of Fortran subroutines for producing Hbook output on graphic devices. It is designed to produce drawings and slides of a quality suitable for talks and publications. Although it does not produce all the numerical information of the Hbook output routines, it is not restricted by the line printer's poor resolution and single character size.

Hplot can access an Hbook data structure and transform it into drawings using the low-level routines of any graphics package, including GD3,⁴ Tektronix Plot 10, and Smog. The drawings can be reproduced on any device available at the computer center, but the preferred output device is the Plotter, for which the character size and position have been optimized. Other options include

- Predefined ISO standard paper sizes, horizontal or vertical orientation, and suitable margins.
- Combination of several graphs on the same page, either by windowing or superimposition, or both, distinguished by different symbols.

- Titles on the axes and text anywhere on the picture, with Greek characters supplementing the several character fonts already implemented.
- 3-D surface representation for scatter plots.
- Color, if the hardware permits it.

The Hbook/Hplot system has already been implemented at several hundred places in the world, in applications extending far beyond the domain of high energy physics.

At the Rutherford Appleton laboratories the package in general use is Smog. It is available from Rutherford Appleton's Atlas division library and is intended primarily for generating output for Information International Inc.'s FR80 microfilm recorder. Smog software generates points, lines, arcs, characters, and text, and accommodates changes in column format, intensity, character length, size, and spacing. The user can select the coordinate system, rotate text, and define and use pictures stored on an FR80 disk. There is also a fast-growing library for high-level routines that use Smog as a low-level system.

GD3,⁴ a device-independent graphics system written at CERN, produces a metafile that can be viewed on a variety of devices for which drivers are available. This package runs on different computer systems and is used in many institutions working in high energy physics.

Graphics in the interactive data analysis stage

At the very end of the processing chain, the results are presented in seminars, conferences, and literature. For that purpose, interactive tools are used to produce drawings as quickly as possible. The systems in use are Sigma, HTV, and IPS.

Sigma, or System for Interactive Graphical Mathematical Applications,⁵ is a programming language for scientific computing that runs on Control Data Corporation Cyber machines under the NOS operating system. Its major characteristics are the following:

- (1) The basic data units are scalars, one-dimensional arrays, and multidimensional rectangular arrays, which Sigma handles automatically.
- (2) The calculational operators of Sigma closely resemble the operations of numerical mathematics; procedural operators are often analogous to those of Fortran.
- (3) The system is designed to be used in interactive mode on terminals connected to a central computer, and can conveniently display arrays in the form of sets of curves.
- (4) The user can construct his own programs within the system, has access to a program library, can store and retrieve data and programs, and obtain alphanumeric and graphical hard copy on request.
- (5) The Sigma implementation is a multiaccess time-sharing system using the central processor only for actual computation. It also provides batch processing, as the user-written code is the same for both interactive and batch modes.

Figure 10. Some optional Hbook output presentations on a line printer.

The system's main drawback is its lack of probability; nevertheless, it has been highly appreciated by theorists.

HTV⁶ is the interactive version of the Hbook/Hplot system. It is written in ANSI Fortran and has been implemented on most of the computer systems on which interactive facilities are available to physicists.

HTV consists of a command interpreter, ZCEDEX,⁷ which handles the keyboard input and provides the following capabilities.

- (1) Creation of macros by means of predefined system commands.
- (2) Modification of existing macros with an editor compatible with the Wylbur editor (used on IBM systems).
- (3) Preservation in a macro library of created macros for use in a later session.
- (4) Creation and manipulation of arrays.

In addition to the ZCEDEX facilities, all Hbook and Hplot utilities are available. For example, "CALL HPLLOT(ID)" in the standard Fortran Hplot program is replaced in HTV by "PLOT ID."

In most cases the objects manipulated by HTV are plots created in one of the previous stages of the analysis. For example, to get a plot from the histogram data base, the user types FETCH ID, in which ID is the plot identifier.

As in the standard Hplot, comments can be added anywhere on the picture in a variety of character sets. In addition to the standard screen output, a display file with only the desired pictures can be produced and sent to the plotter.

IPS,⁸ developed in DESY in Hamburg, uses TSO on the IBM 370/168.

Two software systems are involved in a graphics session under TSO. The first is the Desynet and controller software used to connect a terminal to the IBM. Written in Pascal and Assembler languages, it enables the user to build a "virtual link" to the host, run his session, and ultimately delete the link, using a very simple command language.

The second system resides on the IBM. A new TSO command processor was written that allows the user to activate any general or user-written graphic application module. A subroutine package manages terminal I/O of graphic data and allows for dynamic allocation of data sets while the program is running.

Figures 5, 8, 10, 11, and 12 are examples of plots produced by these systems.

Conclusions

A very large program base has been installed on a wide variety of computers for high energy physics applications, but many programs will have to be redesigned and many others will be added and rewritten to meet the requirements of the future LEP experiments. A growing number of these programs are using graphics, and there is a strong need to standardize such tools as graphics packages, metafiles, and device interfaces. ISO's proposal of a two-dimensional graphics standard is therefore being welcomed as a first step in this direction.

For future experiments much work will have to be invested in the exploitation of interactive graphics. The

Figure 11. This typical Hplot output shows subatomic disintegration spectra. *Ereignisse* are "events"; *Zerfalls* means "decay."

Figure 12. Data obtained from a histogram data base: (c) is the result of the division of (a) by (b); the superimposed curve shows a polynomial fit to the data.

complex LEP experiments in particular will need intelligent graphics workstations with high-resolution color raster screens capable of handling a large and complex graphics data base that includes a variety of viewing operations. ■

References

1. J. Bettels, "MERLIN, a Graphics Facility for the Proton/Antiproton Collider Experiments," presented at the Europhysics Conference on Computing in High Energy and Nuclear Physics, University of Bologna, Sept. 1980.
2. R. Brun, I. Ivanchenko, and P. Palazzi, *HBOOK User Guide*, CERN Program Library, Y250.
3. R. Brun and H. Watkins, *HPLLOT User Guide*, CERN Program Library, Y251.
4. R. Miller, *GD3*, CERN Program Library, J510.
5. R. Hagedorn, J. Reinfelds, C. Vandoni, and L. Van Hove, *SIGMA, a New Language for Array-Oriented Computing*, CERN 78-12.
6. R. Brun, *HTV User Guide*, CERN DD/EE/80-5.
7. R. Brun, *ZCEDEX User Guide*, CERN, DD/EE/80-6.
8. P. K. Shilling, *Computer Graphics at Desy*, DESY DV/79-03.

Jürgen Bettels is a member of the Data Handling Division of the European Organization for Nuclear Research (CERN) in Geneva. Since 1980 he has led a project for using high-performance graphics stations in data analysis for proton-antiproton experiments. His special interests include the exploitation of intelligent, personal graphics workstations and the definition of international graphics standards. He joined CERN in 1971, after receiving a physics degree from the Technische Hochschule Aachen in Germany.

Bettels is a member of ACM, Eurographics, and SCGA and is involved in the ISO's efforts to find and establish graphics standards.

René Brun is a physicist who has worked at the European Organization for Nuclear Research (CERN) in Geneva since 1973. His main task has been to coordinate the software development and the data analysis programs in the NA4 experiment. He will be undertaking a similar task in a LEP experiment run by 200 physicists. One of his main interests is designing general graphics tools that can run on many different computer systems.

Brun graduated from the Clermont-Ferrand University in France, where he obtained his PhD in nuclear physics.