Software Sessions

Introduction to Software

Michael McCullagh
University of Nottingham, U.K.

McCullagh: The United States appears very bright to me compared with the U.K. Last year we had a miner strike which led to complete blackness; practically every industrial organization in the country was dark for 8 hours each day. Once again a miner strike seems likely and lights will be going out again, except in the computing and software sectors where development continues unabated, as it should.

Software is generally divided into two categories—systems software and applications software. Systems software includes driving routines and interactive software that handles, for example, movement of the cursor over an interactive screen. The single characteristic for separating these two categories is that systems software is normally written in a very efficient machine language. We heard yesterday about hardware but not about the software that makes the hardware operate. Although there are many people working in systems software, I regard the task as only a small part of the graphics endeavor. The systems software is also likely to be untransferable between centers. This software is obviously completely hardware dependent, because often the computer goes with the hardware.

Applications software, for data bases, data structures, digital terrain models, manipulation, symbology, and information systems for cartographic display, is a rather different set of programs using high-level language. This language, often FORTRAN, is compatible with most machines in the world today. If the programs are written with transferability in mind, they are probably completely compatible except for one or two random factors which always occur. I hope that all types of application software to be discussed today will be FORTRAN and transferable. One of the big problems, despite the fact that I'm European, is that I don't approve of writing in a software language such as ALGOL because it is applied differently throughout the world. FORTRAN does not tend to have this problem as it is pushed by the largest computer manufacturer.

All the sessions today will consider methods that tend to be style dependent. If you have very little data in your data base, say only a few thousand records, it is not worthwhile to move to a superb, automated structure for retrieval. For terrain analysis, unless you have large sets of data and large files which you want to keep in a big memory, you can store information block by block.

The biggest problem with automated cartographic information systems is that information sources are not often in completely readable form and must be made so. You will usually have a scale problem. And will
the software be used for research or for production? In one situation
you are inundated with data which has to go through the pipeline as
fast as possible, clear to the finished map. In another situation
you will probably spend far longer processing the data to achieve a
specially designed output. Another consideration is the size of the
computer at your disposal.

According to my "plastic-bottle" philosophy, there is an ubiquitous
computer—a small minicomputer system—which practically any firm or
small research organization can afford. Effectively you have a domestic
user for your "plastic bottle." The alternative is the large machine
which can be equated to the SST—undoubtedly worthwhile, but tremen-
dously expensive, big, complex, and always breaking down; you will have
difficulty making it work. In other words, unless you are extremely
involved in software operation, you do not want to be concerned with
making the computer work; you want to be concerned with the cartography.
Perhaps cartographers can now be less dependent on the development
of big machines and manage with much smaller machines which are still
reasonably fast when compared to standards of 2 or 3 yr ago.

My "plastic bottle," which I use at Nottingham, is in my opinion
excellent for research work. It could also be used for production if
you improved the peripherals. I think that you will get more for your
dollar using the "plastic-bottle" approach rather than the SST-type
approach. In terms of graphics, more work (admittedly at a less flashy
level) could be accomplished using these basic small computers. The
"plastic-bottle" approach also seems to have a grass-roots effect—
people who are otherwise unconnected with graphics become more concerned
and aware of what you are trying to do. Automated cartography would
be ubiquitous.

Lastly I want to discuss the accuracy of graphic designs and how
accuracy is affected by different features (in a very broad sense).
Accuracy and repeatability are only required where you are producing
maps to be used by legal arbiters of boundary disputes. For most other
mapping, accuracy of input data is relatively unimportant. The relief
representation will look fine, regardless of whether the data are exact
or only marginally correct; in fact in many cases you cannot get that
accurate data anyway.

(Slide) (Editor's note: Slides not available for publication.) Here
is John Friend's idea of accuracy in the 1700's, a time when they
believed in making maps look pretty. They also believed in making maps
as accurate as possible, and this is the result.

(Slide) Maps produced by the computer can be made to look pretty by
adding color—perhaps not like this but appealing to the domestic
consumer.

(Slide) This was produced in London by Richard Howthe for geochemistry
units—an innovation in using automated cartographic techniques to
represent three variables with three different colors. This type of
cartography is widely criticized but can be used very effectively to
illustrate relationships between variables by making them spatially
obvious.

(Slide) This is the power spectrum of an image that has been contoured.
You want accuracy, and you want to make it look less mundane and less
circular, hence the color.

(Slide) This is a graphic for simply representing data. Fairly high inaccuracy can be tolerated. Very little computer time and programming expertise are required to create this type of display. It does, however, lack the appeal of color.

(Slide) Here is a view of two intersecting wave forms. Although the plotter that produced this was cheap and very wobbly and the line density and representation are not what they might be, the addition of color has improved the impression of accuracy.

To summarize I suggest the use of a small machine for developing software and a series of programs which are only as good as you need. Maybe we could see more innovative representations from automated cartographic techniques.