Multimedia Enhances GIS Applications

*By **Martin Huber**, research coordinator, Laikipia Research Programme (Universities of Berne, Switzerland, and Nairobi, Kenya), PO Box 144, Nanyuki, Kenya.*

An incredible shift is occurring in geographic data dissemination. Spatial data soon will be on everyone's computer — from shopkeeper to minister, sales manager to taxi driver. Most of these people are not used to reading maps, yet geographic information could be vital for their decisions. To support such diverse applications, geographic software must be intuitive and easy to learn. Users should be able to carry out the functions they need and to immediately grasp essential information. Thus, the stage is set for combining multiple software packages, including multimedia.

Extending Multimedia to GIS

Because data and GIS algorithms are in a multitasking computer environment, there are many possibilities for presenting geographic data. Instead of printing maps only specialists can read, messages transmitted via the computer itself can open new perspectives. While a single modern software package cannot solve every problem, it should solve a particular group of problems well and be able to link to other programs that perform other tasks. Such packages have a variety of data and programming interfaces, allowing functions and data to be combined effectively. GIS software and spatial data could drive such a multisoftware-based presentation platform (see figure, page 52).

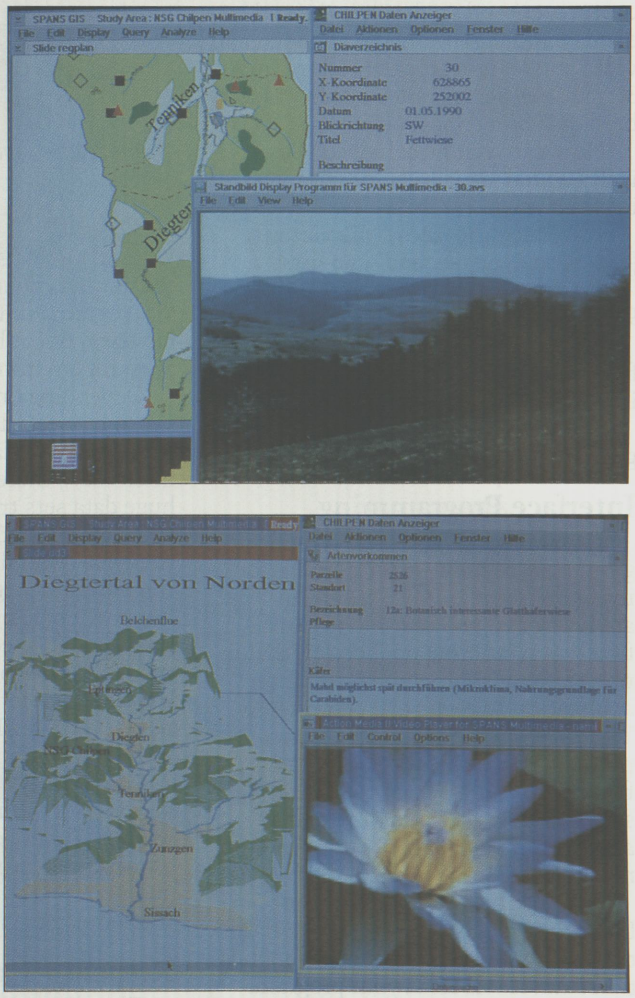
For example, a simple GIS-multimedia application could be developed with a picture-support extension to a GIS's database. Some agencies use GIS as an electronic atlas filled with inventories of spatially distributed objects like nature reserves, archeological sites and historical buildings. Whenever someone applies for a building permit, the inventories have to be consulted to avoid conflicts with protection policies. If there is a possible conflict, an official normally has to visit the site and check the situation in detail. If all the relevant inventories are georeferenced in a GIS, well-described in the database and illustrated with pictures or short video sequences, the potential conflicts can be found quickly and an inspection decision made on the spot. Such a GIS-multimedia database would speed up daily duties to free time for other tasks.

Data Organization

A database management system is the heart of an integrated GIS-multimedia environment, and there are basically two ways to organize data in a multiprogram system. The first approach is to store all data in the database and deliver them to a requesting application in the format specific to that application. The second strategy allows each application to maintain its own filing system, and the database contains only pointers to the respective flat files. The latter approach is pragmatic, because there are only a few database engines on the market that smoothly deal with the binary data types (5MB or larger) needed for some multimedia objects. The former approach is preferred from an organization point of view, because all data related to a subject are tied to that subject — a considerable advantage in backup procedures. The first approach also fits well in a local area network (LAN) environment. There are, however, problems associated with accessing data for display programs:

1. Reading data from disk can be faster than a query-and-read process in a database management system.
2. Transferring huge multimedia files with interprocess communication consumes more system resources than a direct read. Furthermore, a lot of programs are not yet suited for receiving their internal format data through interprocess communication.

A multimedia-GIS prototype recently created for a natural resources protection authority in Switzerland uses the flat file approach with IBM's OS/2 Extended Services Database Manager. Each picture or video object exists only as a file name in the database and, on request, the file name is passed via Dynamic Data Exchange (DDE) to the display program, which fetches the data from disk. On the geographical side, where TYDAC's SPANS GIS software is used, only the information about the location is kept in internal format, but all object descriptions reside in the database. In this small prototype you already have to keep track of three different file types (database, geographical and pictures), and all the pointers have to be entered explicitly. An operative system would need a mechanism to just click on an object to put its pointer into a database field. In short, organizing all data within a database



Multimedia integration into a GIS enriches map information by on-line site or object.

management system will make the GIS-multimedia hybrid more manageable. Moreover, a few database providers intend to integrate multimedia data types into their database and communication concepts. Software developers should think about accessing data for their applications from a common database, rather than from flat files.

Data Compression

Multimedia often is associated with large data sets. Therefore, some people consider the technology to be only usable as a standalone solution on specialized hardware. For some responsible for LANs, multimedia is just another nightmare. Data compression is a must if you want to bring multimedia into working practice. The Swiss prototype system uses IBM's Action Media II (AM II) multimedia adapter, effectively using video compression.

AM II is based on Intel's DVI chip, which is a hardware implementation of some compression and decompression algorithms. The chip can be used to capture, compress, decompress and display audio, still images and video. In video mode, it is able to produce up to 30 pictures per second with an ordinary hard disk as a data source.

How can that be achieved? Basically, video signals can be compressed in three ways:

1. Reduce the size of the picture (e.g., drop every second scan line).
2. Choose a smaller color depth (e.g., 9 bits/pixel instead of 24 bits/pixel).
3. Store only the changes from frame to frame instead of every frame in full size.

AM II combines all of these strategies (Fry, 1991).

Interface Programming

Combining programs in a multiprogram environment requires paths for interprocess communication. An application should be able to run a function of another application that is not active or to instruct a running application to perform a certain task. There are several concepts of interprocess communication. The more flexibility you get from the operating system, the easier it is to design an application environment that fits your needs.

The Application Programming Interface (API) is a low-level interface that allows you to call another application's functions from your program. The program has to be compiled and, at run time, the external function is linked dynamically. Linking a second application is transparent to the user, so there is a smooth interface. The API concept is implemented in the OS/2 operating system. A major problem with APIs, however, is that software developers seldom provide a full interface because of copyright and licensing issues regarding use of the API code.

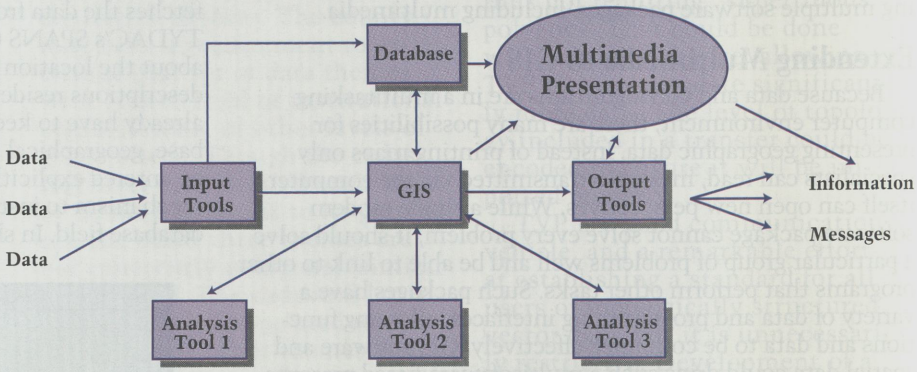
Dynamic Data Exchange (DDE) is a kind of pipe for passing data and commands from one application to another. DDE is relatively easy to implement; it only needs the communication to be initialized and a loop for DDE messages. DDE was used in the prototype, and it was efficient with small data packages like commands. Nevertheless, only a few software packages on the market make extensive use of DDE for program control. That will change, however, once the potential of a multiprogram environment is recognized.

Depending on the data-organization scheme selected (database with pointers to external data or with data stored internally), there are either small or huge data sets transferred from one application to another. Under OS/2 there are several concepts for data transmission, e.g., DDE, named and unnamed pipes, and queues or shared memory. To speed up overall performance, you also can create buffer routines that fetch data before a user requests them. That task can be performed when no interactive job is working. Data flow is crucial for the performance, but it cannot be explained in full detail in this article. It is worth considering several alternatives and making tests with real data before implementing a final version.

Response Time

The system should respond to every request reasonably quickly to avoid annoying the user. That is not easy if several data-intensive jobs run simultaneously on the same machine. OS/2 provides a preemptive multitasking environment that changes job priorities dynamically. The strategy gives the highest priority to the application the user interactively works with. If all programs involved in the GIS-multimedia system are not changing priorities individually, the operating system will optimize system resource distribution.

The prototype implemented on OS/2 responds fairly quickly. On an ordinary '486 33MHz PC, a user can query points on a map while a video sequence is running, and as soon as another video sequence connects to the queried point, the old video is interrupted and the new one begins to play.



An integrated, multiapplication environment with multimedia can use geographical data to drive information transfer and decision support.

A New Approach

Multimedia will change the way people look at maps in a GIS. Moreover, the technology will help people become more productive. Setting up such systems will entail major organizational and technical problems, but the technology is progressing quickly and solutions are under way. Modern operating systems make it possible to run complex application hybrids even in a PC environment. PCs easily can be used, therefore, as front-end machines for analyses, queries and presentations in a mainframe- or workstation-based custodial GIS. An integrated GIS strategy that includes multimedia presentation will add even more value to geographic data collected on a long-term basis.

Reference

Fry, T. 1991. *Action Media II Application Programmer's Guide.* IBM Mechanicsburg/Winchester.