C H A P T E R 8

How to Pick a GIS

8.1 THE EVOLUTION OF GIS SOFTWARE
8.2 GIS AND OPERATING SYSTEMS
8.3 GIS FUNCTIONAL CAPABILITIES
8.4 GIS SOFTWARE AND DATA STRUCTURES
8.5 CHOOSING THE "BEST" GIS
8.6 STUDY GUIDE
8.7 EXERCISES
8.8 REFERENCES
8.9 KEY TERMS AND DEFINITIONS

Was there ever in anyone's life span a point free in time, devoid of memory, a night when choice was any more than the sum of all the choices gone before?
—Joan Didion, Run River, *1963*

You pays your money and you takes your choice.
—Punch, 1846

"Can't act. Can't sing. Balding. Can dance a little."
—MGM executive report on Fred Astaire's screen test

"That's all well and good, bud, but does it come with a remote?"

8.1 THE EVOLUTION OF GIS SOFTWARE

One of the first tasks a GIS user faces is deciding which GIS software to use. Even if a GIS has already been purchased, installed, and placed right in front of your nose, it is very natural to wonder whether some other GIS system might be better, faster, easier to use, have clearer documentation, or be better suited to the actual task you are working on. This chapter gives some of the background necessary to make an intelligent GIS selection. There is quite a history to learn from, including some excellent accounts of spectacular failures, but also many examples of clear statements of how things went right. Examples from the early days of GIS are the papers by Tomlinson and Boyle (1981) and Day (1981). The philosophy here is that the educated consumer is the best GIS user, and an effective user soon becomes an advocate and sometimes a GIS evangelist. This chapter is not intended to tell you which GIS to buy or use. Rather, it is hoped that, it will help you to decide this for yourself.

As is often the case, a good education begins with a little history. Chapter 1 introduced overall GIS development in terms of the distant origins of geographic information science as a whole. This was difficult to do without mentioning specific GIS software packages. Now it is appropriate to discuss the development of software in more detail.

8.1.1 The Ancestors of GIS Software

GIS software did not suddenly appear as if by magic. There was a lengthy period leading up to the first real GISs during which the breed evolved rather rapidly. As we saw in Chapter 1, the intellectual ancestry included the creation of a spatial analysis tradition in geography, the quantitative revolution, and dramatic technological and conceptual improvements in the discipline of cartography.

An early GIS landmark was an international survey of software conducted by the International Geographical Congress in 1979 (Marble, 1980). This survey had three volumes, one of which was entitled *Complete Geographic Information Systems*, although in fact few true GIS packages were represented. This volume was influential in deciding on the name "GIS" because many alternatives were in use at that time. Just as important were the two volumes *Cartography and Graphics* and *Data Manipulation Programs*. Together, these three volumes encapsulated the state of geographic data processing in the 1970s (Brassel, 1977). Most cartographic programs were single-purpose FORTRAN programs to do individual GIS operations such as digitizing, data format conversion, plotting on a specific hardware device such as a pen plotter, map projection transformations, or statistical analysis of data. None of these packages were integrated; a typical use would be to apply a series of one-at-a-time geographic operations to arrive at a final result or map.

Some of the early computer mapping systems had already devised many GIS functions by this time, however. Among these were SURFACE II by the Kansas Geological Survey, which could do point-to-grid conversions, interpolation, surface subtraction, and surface and contour mapping; CALFORM, a package that could produce thematic maps; SYMAP, a sophisticated analytical package from the Harvard Laboratory for Computer Graphics and Spatial Analysis that nevertheless ran only on mainframe computers and gave line-printer plots; and the Central Intelligence Agency's CAM, which made plots from the World Data Bank outline maps with different map projections and features.

By 1980 the first computer spreadsheet programs had arrived, led by the VisiCalc program, a very early microcomputer software "killer app." VisiCalc contained only a few of the capabilities of today's equivalent packages, yet for the first time gave the ability to store, manage, and manipulate numbers in a simple manner. Above all, data could be seen as active in a spreadsheet rather than as a static "report" that consisted of a pile of computer printout. The links to statistical graphics, now common in packages such as SASGRAPH and Harvard Graphics, were a natural extension of this capability.

The ancestry of GIS is completed by the first advances in database management systems. Early systems for database management were based on the less sophisticated data models of the hierarchical and related data models. A landmark was the beginning of the relational database managers in the early 1970s. Relational database managers quickly became the industry standard, first in the commercial world of records management and later in the microcomputer world.

8.1.2 The Early GISs

By the late 1970s all of the necessary parts of a GIS existed as isolated software programs. The largest gap to be filled was between the relational database manager and the programs that dealt with plotting maps. The specific demands of hardware devices from particular manufacturers kept this as a constantly evolving field, with frequent rewrites and updates as systems and hardware changed. Later, the device independence attributable to common operating systems such as Unix and computer graphics programming standards such as GKS, Core, and PHIGS led to a narrowing of this chasm, to the point where today it remains as barely a discernible dip in the GIS ground. The scene was set for the arrival of the first true GISs.

As we saw in Chapter 1, one of the earliest civilian systems to evolve all the capabilities of a true GIS was the CGIS (Canadian Geographical Information System), mostly because this system was the first to evolve from an inventory system toward doing analyses and then management. Essential to the emergence were the georeferencing and geocoding of the data, database management capability, a single integrated software package without separate, stand-alone elements, and a single user interface.

At first, GIS packages had unsophisticated user interfaces, and many actually made the user write short computer program–like scripts or to type highly structured formatted commands one at a time into the computer in response to prompts. As the GIS software evolved, the need for upward compatibility—that is, the need for existing users to be satisfied with a new version because things still work in much the same way as before—meant that many systems preserved elements of these older user interfaces long after they had been replaced by better tools.

The second generation of GIS software included graphical user interfaces, usually involving the use of windows, icons, menus, and pointers. In the typical configuration today, the windows are standardized by the operating system and function in the same way that it does, "inheriting" its characteristics. A first generation of GIS software used windows custom-built by the vendor. Later, after the broad distribution of windowing systems such as X-Windows and Microsoft Windows, the graphical user interface (GUI) tools that are part of the operating system became accessible to software designers and programmers.

The typical system has pop-up, pull-down, and pull-right menus for selecting choices. Choices and locations are indicated with a mouse, although some systems use track balls or light pens. Similarly, the typical GIS can support multiple windows—for example, one for the database and one to display a map—and the tasks can be opened and closed as needed. While closed, they function in the background while they are graphically represented on the screen as an icon or small picture.

8.2 GIS AND OPERATING SYSTEMS

Early GIS was heavily influenced by the types of operating systems in use. Early operating systems were quite unsophisticated but were used with GIS nevertheless. Among these were IBM's mainframe operating systems, MSDOS by Microsoft, and DEC's VMS. These were rapidly replaced as the various GUI-based operating systems came into operation and as the microcomputer and workstation took over from the minicomputer and mainframe.

In the microcomputer environment, the GUI-based operating systems include Windows, Windows-NT, and Windows 95. The unified user interface, revolutionized by the Apple Macintosh's GUI and desktop metaphor, quickly took over as the dominant microcomputer operating environment, although others, such as IBM's OS/2, have remained popular also. These operating systems added two critical elements to the microcomputer's capabilities: *multitasking* (allowing many simultaneous work sessions) and *device independence*, meaning that plotters and printers could be taken out and assigned to the operating system instead of the GIS package, in somewhat the way that printing and screen fonts are handled centrally, rather than duplicated in every Windows package.

One system that had encompassed these capabilities since its inception, and that swept the workstation environment, was Unix. Unix is a very small and efficient central operating system that is highly portable across computer systems. It has been the dominant workstation environment for two reasons: first, because it has complete integrated network support, and second, because several full GUIs exist for Unix in the public domain, the most important being the X-Windows system. X-Windows implementations of most leading GUIs exist, including OpenLook and the Open Software Foundation's MOTIF interface. In many Unix systems, the user can switch the GUI to suit particular needs or applications. A full GUI programming tool kit, including such tools as Xt, Xview, and the X-Windows libraries Xlib, is part of the X-Windows release.

As a final benefit, several versions of Unix and all of the GUI systems run extremely efficiently on microcomputers, including shareware Unix releases such as Linux, not only outperforming the Windows-type GUIs, but being available free or as shareware on the Internet or from inexpensive suppliers on CD-ROM. A key element here has been the Free Software Foundation's releases, including GNU (GNU is NOT UNIX) versions of virtually every key element of Unix.

Thus, two main avenues for GISs have evolved as far as operating systems are concerned. On the microcomputer platform a lingering set of DOS applications is rapidly being rewritten for the updated versions of Microsoft's Windows. In this GIS environment, the number of systems installed, the mobility of laptop and subnotebook computing, and the low cost of software have been major strengths. On the workstation platform, Unix and X-Windows, often with MOTIF as the GUI, reign supreme. This work environment has led to high-end applications, large data sets, networking, depth of software, and high-quality graphics. Both are healthy and prospering workplaces for GIS.

8.3 GIS FUNCTIONAL CAPABILITIES

A GIS is often defined not for what it is but for what it can do. This *functional definition* of GIS is very revealing about GIS use, because it shows us the set of capabilities that a GIS is expected to have. A minimal set of capabilities can be outlined and each GIS package held up to see whether it qualifies. A thorough examination of GIS capabilities is the critical step in how to pick a GIS, because if the GIS does not match the requirements for a problem, no GIS solution will be forthcoming. In contrast, if the GIS has a large number of functions, the system may be too sophisticated or elaborate for the problem at hand—a sledgehammer to crack a nut.

The functional capabilities can be grouped by the categories we have used in this book, closely following the Dueker definition of GIS. These are capabilities for data capture, data storage, data management, data retrieval, data analysis, and data display. These "critical six" functions must always be present for the software to qualify as a GIS. We examine each of these in turn.

8.3.1 Data Capture

As we saw in Chapter 4, getting the map into the computer is a critical first step in GIS. Geocoding must include at least the *input* of scanned or digitized maps in some appropriate format. The system should be able to absorb data in a variety of formats, not just in the native format of the particular GIS. For example, an outline map may be available as an AutoCAD DXF format file. The GIS should at a minimum be capable of absorbing the DXF file without further modification. Similarly, attributes may already be stored in standard database format (DBF) and should be absorbable either directly or through the generic ASCII format.

Before a map can be digitized, however, it needs to be prepared. Different GIS packages handle the amount of preparation required in quite different ways. If the package supports scanning, the map needs to be clean, fold-free, free of handwritten annotation and marks, and on a stable base such as Mylar. If the map is digitized by hand it may need to be cut and spliced if the package does not support *mosaicing* (Figure 8.1), and control points with known locations and coordinates need to be marked for registering the map onto the digitizing tablet. Some GIS packages have extensive support for *digitizing* and sophisticated *editing* systems for detecting and eliminating digitizing errors. Others have few or none.

We also saw in Chapter 3 how essential it was to edit the maps after they have been captured. This requires the software to have an editing package or module of some kind. For a vector data set, at the minimum we should be able to delete and reenter a point or line. For a raster, we should be able to modify the grid by selecting subsets, changing the grid spacing, or changing a specific erroneous grid value.

Other functions typical of an editor are *node snapping*, in which points that are close to each other and that should indeed be the same point, such as the endpoints

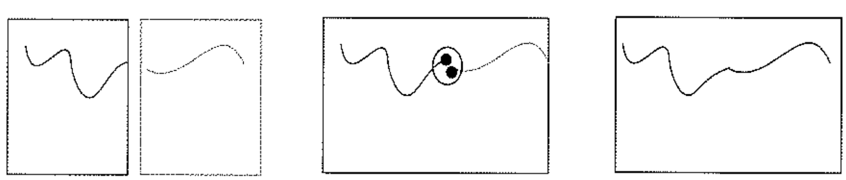


FIGURE 8.1: Steps in mosaicing. Left: Two maps show one feature, but there is a gap. Center: Map edge is merged; nodes are snapped to "zip" feature. Right: Mosaiced map with continuous feature and dissolved map edge.

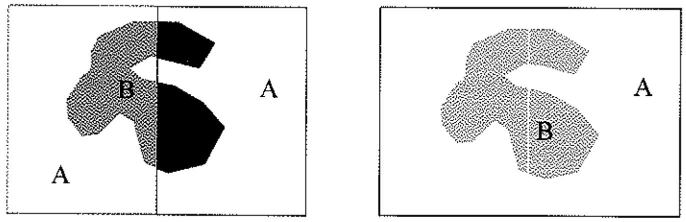


FIGURE 8.2: Steps in the dissolve operation. Left: Two maps show one feature, split across an edge. Attribute and graphic database have three records for type "B". Right: After dissolve, edge lines are removed and the three type "B" records are amalgamated.

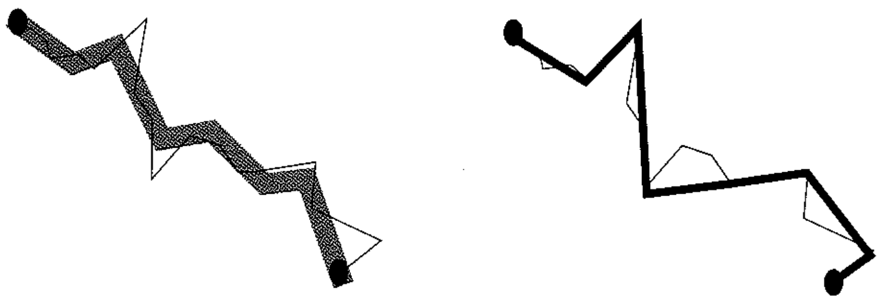


FIGURE 8.3: Map generalization includes eliminating all points along a line falling within a band or buffer around a simplification of the line, with a width sometimes called the "fuzzy tolerance." This way, minor errors such as sliver polygons can be eliminated when data have been captured from maps at different scales.

of a line segment, are automatically placed into the graphic database with the identical coordinates; *dissolve*, when duplicate boundaries or unnecessary lines (e.g., the digitized edges of adjacent category-type maps) are eliminated automatically or manually; and *mosaicing* or "zipping," in which adjacent map sheets scanned or digitized separately are merged into a seamless database without the unnecessary discontinuities caused by the lack of edge matching of the paper maps (Figure 8.2). For example, a major road that crosses two map sheets does not need to be represented as two separated features in the final GIS database.

Another important editing function is the ability to deal with map *generalization*. Many digitizing modules of GIS systems, and certainly scanning, generate far more points than are necessary for the use of the GIS. This extra detail can complicate data reformatting and display, slow the analysis process, and lead to memory problems on the computer. Many GIS packages allow the user to select how much detail to retain in a feature. Most will retain points that have a minimum separation and snap together all points within a fuzzy tolerance (Figure 8.3).

For point data sets, most GIS packages will eliminate or average duplicate points with the same coordinates. Some will allow *line generalization*, using any one of many algorithms that reduce the number of points in a line. Common methods include extracting every *n*th point along the line (where *n* can be 2, 3, etc.) according to the amount of generalization required, and Douglas–Peucker point elimination, which uses a displacement orthogonal to the line to decide whether a point should be retained (Figure 8.4).

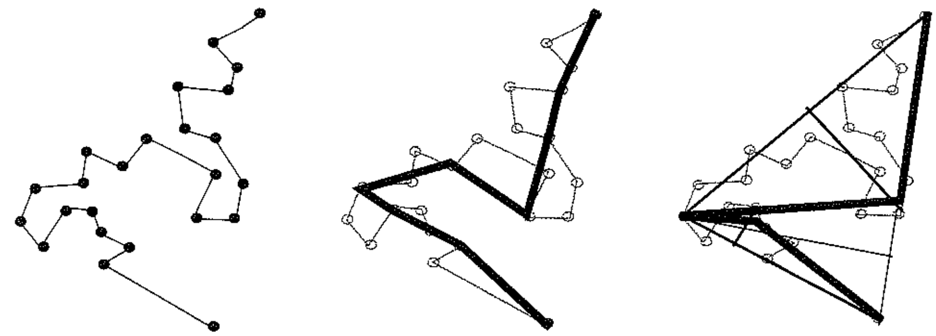


FIGURE 8.4: Line generalization alternatives. The line (left) can be resampled by retaining every nth point (center), or by repeatedly selecting the most distant point from a line between end nodes (right) and redividing the line until a minimum distance is reached, the Douglas–Peucker method.

Area features can be eliminated if they become too small, or can be grouped together, a process many GIS packages call *clumping*. It is also possible to generalize in the attributes, joining classes together, for example.

To be useful, a GIS must provide tools above and beyond the editor to check the characteristics of the database. Checking the attributes is the responsibility of the database manager. The database system should enforce the restrictions on the GIS that are specified during the data definition phase of database construction and stored in the data dictionary. Most of this checking is done at data-entry time. It checks to determine that values fall within the correct type and range (a percentage numerical attribute, for example, should not contain a text string and should have a record of less than or equal to 100).

More intricate and demanding are checks on the map data. Some GIS packages, which do not support topological structuring, do not enforce any restrictions on the map. Some simply check ranges; for example, every grid cell should have a data value between 0 and 255 in an image map. These systems run the risk of lacking a match between the attributes and the space they represent. No part of the map, for example, should fall into two separate areas—that is, the areas on a polygon map should not overlap or leave gaps. This happens when maps are captured at different scales or from inaccurate sources.

Topological GIS systems can check automatically to ensure that the lines meet at nodes and that the entire map area is covered by polygons without gaps or overlaps. Beyond simply checking, many GIS packages allow automatic cleaning of topology, snapping nodes, eliminating duplicate lines, closing polygons, and eliminating slivers. Some systems simply point out the errors and ask the user to eliminate them with the editor. Some go ahead and make the corrections without user intervention. The GIS user should be careful when using automatic cleaning, for the tolerances may eliminate important small features or move the features around in geographic space without accountability. A specific GIS package may or may not be able to deal directly with GPS data conversion, with survey-type data from COGO (coordinate geometry) systems, or with remotely sensed imagery. Some GIS packages have both functions—that is, they serve as GIS and image processing systems. Among these are Idrisi, GRASS, and ERDAS.

Essential to geocoding capabilities, because GIS allows maps from many sources to be brought into a common reference frame and to be overlaid, is the geocoding software's ability to move between coordinate systems and map projections. Most GIS packages

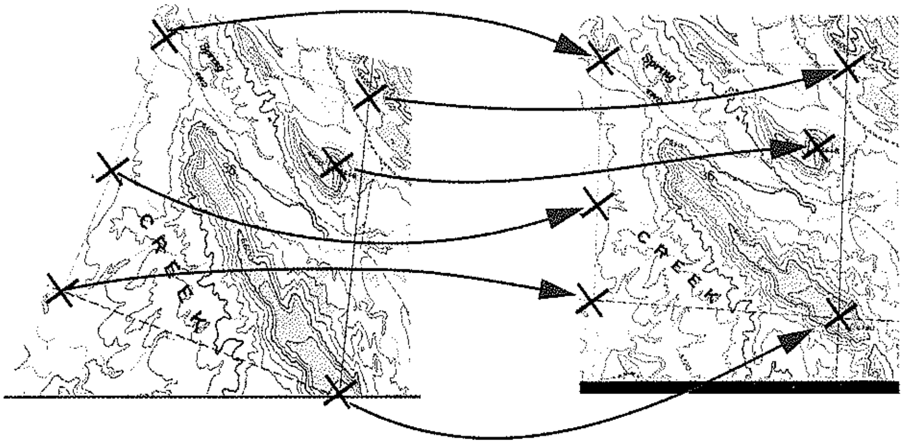


FIGURE 8.5: The rubber sheeting method. A map with unknown geometry (say an air photo or scanned map) can be distorted so that its geometry matches that of another map. Pairs of points must be available both on the image and on the map showing the same place or feature location, called control points. Within the GIS, rubber sheeting warps the geometry statistically into that of the map, so that the two geometries match.

accomplish this using *affine transformations*. Affine operations are plane geometry; they manipulate the coordinates themselves by scaling the axes, rotating the map, and moving the coordinate system's origin.

In some cases, when no good control is available, maps must be statistically registered together, especially when one layer is a map and one an image or photograph. The statistical method known as *rubber sheeting* or *warping* is used for this and is a function inside many GIS packages (Figure 8.5).

8.3.2 Data Storage

Data storage within a GIS has historically been an issue of both space—usually how much disk space the system requires—and access, or how flexible a GIS is in terms of making the data available for use. The massive reductions in the cost of disk storage, new high-density storage media such as the CD-ROM, and the integration of compression methods into common operating systems have made the former less critical and the latter more so.

Current emphasis, therefore, is upon factors that improve data access. This has been a consequence also of the rise of distributed processing, the Internet, and the World Wide Web. As a result, many GIS packages are now capable of using *metadata*, or data about data, in an integrated manner. Metadata support might include a system for managing a single project as a separate entity, to managing many projects with multiple versions, to full support for exchangeable metadata stored in common formats and searchable through online "clearinghouses." The USGS's Global Land Information System, NASA's master directory, and the Federal Geographic Data Committee's Spatial Data Clearinghouse are all examples. Participation in the common library entails both standardizing the metadata to make it searchable and agreeing to make the data available either on or offline.

Other larger issues around GIS use, most essential to the degree of user friendliness of the system, concern the mechanism for user interaction with the software's functionality. Virtually all GIS software allows user interaction via command lines and/or windows within a GUI. The GUI interface is tedious, however, without some way of "*batching*" commands so that they can be executed either at another time, as a background task while the user gets on with another job, or for design-loop editing to change minor aspects of the process. Most systems, therefore, also contain a "language" for the user to communicate with the system. This allows users to add their own custom functions, automate repetitive tasks, and add features to existing modules. These languages are usually command-line programs or *macros*, but they can also be enhancements of existing programming languages such as Basic and Smalltalk.

Although disk storage is less critical than in the past, it can still be a constraint. GIS software on a microcomputer can occupy tens of megabytes even without data, and on a workstation perhaps hundreds of megabytes. As data become higher resolution, as more raster layers are used, and as finer and finer detail becomes available, many GIS data sets can easily move into the gigabyte range in size.

This implies that not only is supporting multiple resolutions important—for example, using coarse browse images as samples of the real thing—but also that data compression should be supported. This can vary all the way from partitioning data sets to meet constraints (such as a maximum number of polygons) to supporting compressed data formats and structures such as JPEG, run-length encoding, or quadtrees.

Also of great importance from a user perspective is the degree to which the system itself provides help to users, either via the operating system or as part of the software. Integration with online manuals, such as in Unix versions, support for context-sensitive hypertext help systems, such as the Windows help feature, and, ideally, an online interactive hypertext help system can be critical for the new user. These help systems can be used only when needed rather than encumbering the advanced user with unnecessary basic information.

Support for data formats is important to a GIS when data are to be brought in from outside (e.g., public-domain data from the Internet). Ideally, the GIS software should be able to read common data formats for both raster (DEM, GIF, TIFF, JPEG, Encapsulated PostScript) and vector (TIGER, HPGL, DXF, PostScript, DLG). Some GIS packages have import functions only into a single data structure, usually either an entity-by-entity structure or a topological structure.

For three-dimensional data, these systems usually support only the triangular irregular network. Others support only raster structures based on the grid, including the quadtree, and either convert all data into this structure or just ignore it. Some GIS packages continue to support only data in a proprietary format, available only at cost from the software vendor. A rather critical GIS function is the ability to convert between raster and vector data, an absolutely essential feature for the integration of multiple data sources such as GPS data and satellite images.

Of increasing interest in recent years has been the development of GIS functions that support data in standard exchange formats. At the national and international levels, several data transfer standards have now been developed, such as the Spatial Data Transfer standard and DIGEST. As these standards become mandated, and as the role of data exchange increases, led by the Internet, most GIS systems will develop support for inputting and outputting data in these standard formats. The 2000 census, with its support for the federal information processing standard for data exchange (FIPS 173), will probably drive GIS vendors to support this necessary next step for GIS.

8.3.3 Data Management

Much of the power of GIS software comes from the ability to manage not just map data but also attribute data. Every GIS is built around the software capabilities of a database management system (DBMS), a suite of software capable of storing, retrieving selectively, and reorganizing attribute information. The database manager allows us to think that all the data are available, that the data are structured in a simple flat-file format, and that they constitute a single entity. In fact, the database manager may have partitioned the data between files and memory locations and may have structured it in any one of several formats and physical data models.

A database manager is capable of many functions. Typically, a DBMS allows data entry, and data editing, and it supports tabular and other list types of output, sometimes independent of the GIS. Retrieval functions always include the ability to select certain attributes and records based on their values. For example, we can start with a U.S. database, and select out all records for states containing cities with over 1 million inhabitants, forming a new database that is wholly enclosed by the original and that duplicates part of it. We can also perform functions such as sorting data by value, and retrieving a selected record by its identification, such as a name or a number.

Address matching involves taking a listed street address, such as "123 Main Street," and using the GIS's existing data to match the address with a geographic region in the GIS. The key to this capability is usually the TIGER files from the U.S. Census Bureau, which contain a topologically connected street and block network, referenced to house numbers. The address match finds the street and then moves along the street's individual blocks until the house number lies within the block and on the correct side of the street.

Many operations on data are very important from a mapping perspective. For example, very often maps captured from different sheets must be merged together, or sometimes a *mask* must be placed over the data to exclude features entirely from the GIS. Examples of masks are private lands within national parks, water bodies, or military bases. Similarly, sometimes data must be assembled in one way, by topographic quadrangle, and then *cookie cut* into another region such as a state or a city boundary. Even more complex, sometimes line features such as the latitude/longitude grid, a river, or a political boundary must be sectioned up or have points added as new features or layers are introduced. This feature, called *dynamic segmentation*, can be done automatically by the GIS (Figure 8.6).

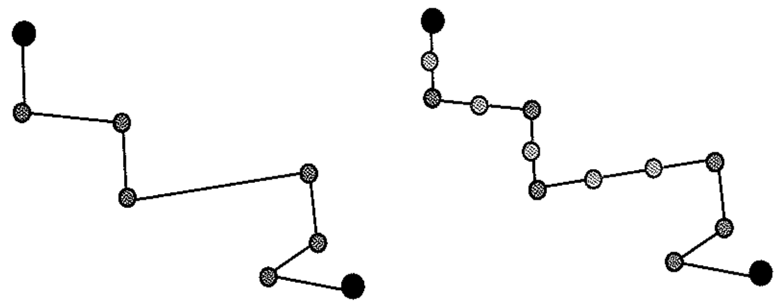
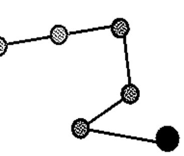


FIGURE 8.6: Using dynamic segmentation, the GIS can create as many segments along a feature as are necessary for analysis or display by adding new nodes (shown in magenta). Each new segment can have its own attributes. For example, it may be necessary to establish a new point to mark every mile measured along a river, and to attach river flow, or pollution data, to the points.

o manage not just map
ftware capabilities of a
ble of storing, retrieving
e manager allows us to
a simple flat-file format,
ger may have partitioned
uctured it in any one of

ly, a DBMS allows data
es of output, sometimes
ability to select certain
e can start with a U.S.
ith over 1 million inhab-
iginal and that duplicates
y value, and retrieving a
er.
ch as "123 Main Street,"
geographic region in the
the U.S. Census Bureau,
ork, referenced to house
ng the street's individual
correct side of the street.
apping perspective. For
t be merged together, or
res entirely from the GIS.
bodies, or military bases.
ographic quadrangle, and
ary. Even more complex,
er, or a political boundary
ayers are introduced. This
by the GIS (Figure 8.6).



egments along a feature as are
Each new segment can have its
nark every mile measured along

8.3.4 Data Retrieval

Another major area of GIS functionality is that of data retrieval. As we saw in Chapter 5, a GIS supports the retrieval of features by both their attributes and their spatial characteristics. All GIS systems allow users to retrieve data—they wouldn't qualify as a GIS if they did not! Nevertheless, among systems some major differences exist between the type and sophistication of GIS functionality for data retrieval.

The most basic act of data retrieval for a GIS is to show the position of a single feature. This can be by retrieving coordinates as though they were attributes, or more commonly by displaying a feature in its spatial context on a map with respect to a grid or other features. For line features, the same goes, with the exception that line features have the attribute of length, and polygon features have the attribute of area. The GIS should be able to calculate and store these important basic properties as new attributes in the database. For example, for a set of counties we may want to take a polygon attribute such as an area of forest and divide it by the county area to make a percentage density of forest cover. Another common measurement we may want is to count features. For example, with the same database we could count the number of fire stations within the same counties by doing a point-in-polygon count from a separate database of municipal utilities and then relate the forest cover to the fire-prevention capabilities.

We have seen that a GIS has the critical capability of allowing the retrieval of features from the database using the map as the query vehicle. One way, indeed the most basic way, of doing this is to support the ability to point at a feature, using a device such as a mouse or a digitizer cursor, to see a list of attributes for that feature. Again, the ability to select by pointing to a location virtually defines a GIS. If it cannot do this, the system is probably a computer mapping system, not a GIS. Just as critical is the database manager *select-by-attribute* capability. This is normally a command to the database query language that generates a subset of the original data set. For example, we could find all houses in a real estate GIS that had been listed on the market in the last year. Similarly, we could find all houses built after 1990. All GIS systems and all database managers support this capability.

As we saw in Chapter 5, GISs allow a set of retrieval operations based on using one or more map features as handles to select attributes of those features. Although some of them are very simple, these operations are also a real litmus test for establishing whether or not a software package is a GIS. A GIS should allow the user to select a feature by its proximity to a point, a line, or an area. For a point, this means selecting all features within a certain radius. For a line or a polygon, we have used the term *buffering*. Buffering allows the GIS users to retrieve features that lie within perhaps 1 mile of an address, within 1 kilometer of a river, or within 500 meters of a lake (Figure 8.7). Similarly, weighted buffering allows us to choose a nonuniform weighting of features within the buffer, favoring close-by instead of distant points, for example.

The next form of spatial retrieval is map *overlay*, when sets of irregular, nonoverlapping regions are merged to form a new set of geographic regions that the two initial sets share. In the new attribute database it is possible to search by either set of units. A GIS should be able to perform overlay as a retrieval operation since to support the many spatial analyses based on map combination and weighted layer solutions, as discussed in Chapter 6. Vector systems usually compute a new set of polygons by adding points to and breaking up the existing sets, and in raster systems we allow *map algebra*, direct addition or multiplication of attributes stored in cells. Map overlay is an important part

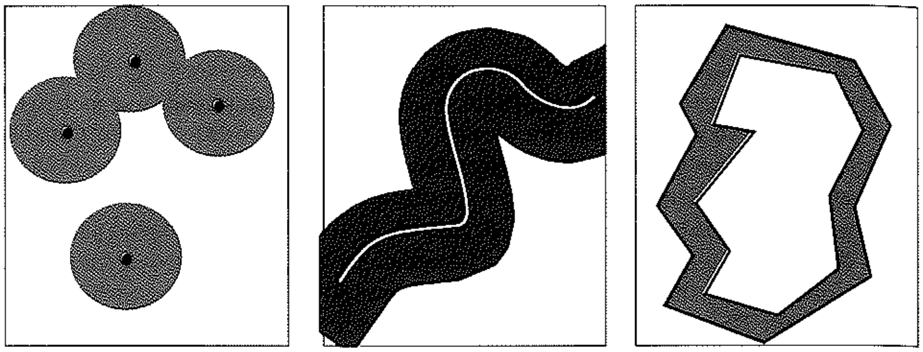


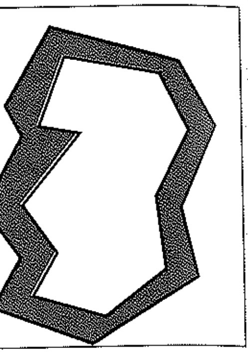
FIGURE 8.7: Buffers can be created around points (left), lines (center), or areas (right). Buffers can be set to a specific distance, such as 1 kilometer, or made a continuous layer of distances from each point on the map to the feature or features in question.

of a major GIS function, that of redistricting, in which new districts can be drawn and the data restructured into the regions so that tests and analyses can be performed by trial and error—for example, to see whether the new districts conform to the federal Voting Rights Act.

Another important set of retrieval options, especially in facilities mapping and hydrological systems, are those that allow networks to be constructed and queried. Typical networks are subway systems, pipes, power lines, and river systems. Retrieval operations involve searching for segments or nodes, adding or deleting nodes, redirecting flows, and routing. Not all GIS systems need these functions, but if the purpose is to manage a system usually abstracted as a network, such as a highway or rail system, a power supply system, or a service delivery system, obviously the GIS should then have this feature.

Dana Tomlin (1990) has elegantly classified the operations that a raster GIS can perform into a structure called *map algebra*. In map algebra, the retrieval operations used are Boolean, multiply, recode, and algebra. *Boolean* operations are binary combinations. For example, we can take two maps, each divided into two attribute codes "good" and "bad" and find a binary AND solutions layer where both layers are "good" (Figure 8.8). *Multiply* allows two layers to be multiplied together—for example, two sets of weights to be combined. In *recode* operations a range of computed attribute codes can be reorganized. An example is taking percentages and converting them to a binary layer by making all values greater than 70% a "1" and all else a "0." Map *algebra* allows compute operations, such as map-to-map multiplication for a binary AND over the space of a grid.

Two truly spatial retrieval operations are the ability to *clump* or aggregate areas, and to *sift*. For example, all areas of saturated soils surrounding swamps could be added to the swamps and recoded as wetlands, making a new, broader category of attribute. Sifting simply eliminates all areas that are too small, individual cells falling between two larger areas, or a tiny sliver polygon. Finally, some complex retrieval operations require the GIS to be able to compute numbers that describe shape. Common shape values are the length of the perimeter of a polygon squared, divided by its area, or the length of a line divided by the straight-line distance between the two endpoints.



as (right). Buffers can be set to
es from each point on the map

istricts can be drawn and
can be performed by trial
orm to the federal Voting

n facilities mapping and
tructed and queried. Typ-
r systems. Retrieval oper-
eleting nodes, redirecting
, but if the purpose is to
highway or rail system, a
the GIS should then have

ons that a raster GIS can
e retrieval operations used
s are binary combinations.
ttribute codes "good" and
s are "good" (Figure 8.8).
ample, two sets of weights
ttribute codes can be reor-
hem to a binary layer by
Map *algebra* allows com-
ry AND over the space of

clump or aggregate areas,
ng swamps could be added
ader category of attribute.
l cells falling between two
retrieval operations require
Common shape values are
its area, or the length of a
dpoints.

1	1	0	0	0
1	1	1	0	0
0	1	0	1	0
0	0	0	0	0
0	0	0	0	0

AND

0	0	0	0	0
0	1	0	1	1
0	1	1	1	1
0	0	0	1	1
0	0	0	0	0

=

0	0	0	0	0
0	1	0	0	0
0	1	0	1	0
0	0	0	0	0
0	0	0	0	0

FIGURE 8.8: Map algebra in its simplest form: Two binary images are ANDed together to give a common area of overlap. Many other operations are possible, such as add, multiply, divide, select maximum, eliminate isolated values, etc.

8.3.5 Data Analysis

The analysis capabilities of GIS systems vary remarkably. Among the multitude of features that GIS systems offer are the computation of the slope and direction of slope (aspect) on a surface such as terrain; interpolation of missing or intermediate values; line-of-sight calculations on a surface; the incorporation of special break or skeleton lines into a surface; finding the optimal path through a network or a landscape; and the computations necessary to calculate the amount of material that must be moved during cut-and-fill operations such as road construction.

Almost unique to GIS, and entirely absent in other types of information systems, are geometric tests. These can be absolutely fundamental to building a GIS in the first case. These are described by their dimensions, point-in-polygon, line-in-polygon, and point-to-line distance. The first, point-in-polygon, is how a point database such as a geocoded set of point samples is referenced into regions. Thus a set of locations for soil samples, generated at random, could be point-in-polygon merged with a digitized set of district boundaries so that a sample list can be sent to each soil district manager. Other more complex analytical operations include partitioning a surface into regions, perhaps using the locations of known points to form proximal regions or Voronoi polygons, or by dividing a surface into automatically delineated drainage basins.

Some of the most critical analytical operations are often the simplest. A GIS should be able to do spreadsheet and database tasks, compute a new attribute, generate a printed report or summarize a statistical description, and do at least simple statistical operations such as computing means and variance, performing significant testing, and plotting residuals.

8.3.6 Data Display

Most of the display capabilities of GISs have been covered in Chapter 7. GIS systems need to be able to perform what has become called *desktop mapping*, generating geographical and thematic maps so that they can be integrated with other functions. GISs typically can create several types of thematic mapping, including choropleth and proportional symbol maps; and they can draw isoline and cross-sectional diagrams when the data are three dimensional.

Almost all GIS packages now either allow interactive modification of map elements—moving and resizing titles and legends—or allow their output to be exported into a package that has these capabilities, such as Adobe Illustrator or CorelDraw. A very limited few GIS packages include cartographic design help in their editing of graphics, defaulting to suitable color schemes, or notifying the user if an inappropriate map type is being used for the data. This would be a desirable feature for many of the GISs on today's market and could avoid many tasteless or erroneous maps before they were created.

8.4 GIS SOFTWARE AND DATA STRUCTURES

In the preceding discussion, the focus was on what functional capabilities the typical GIS offers. It should not be forgotten that many GIS features are predetermined by the GIS's particular data structure. As we saw in Chapter 4, at the very least the underlying data structure that the GIS uses, typically raster or vector but potentially also TIN, quadtree, or another model, such as object-based, determines what the GIS can and cannot do, how operations take place, and what level of error is involved.

In general, the driving force for the choice of structure should be not only what type of system can be afforded, but more critically, what model is most suitable to a particular application, what retrieval and analysis functions will be used most, and what is the acceptable level of resolution and error.

Some examples where particular structures are favored include extensive land characterization applications such as forestry, where detailed data are not required (favors raster); applications involving irregular polygons and boundary lines, such as political units or census tracts (favors vector); applications that require the ability to register all features accurately to ground locations (favors vector); applications making extensive use of satellite or terrain data (favors raster); or applications where image processing functions and analyses such as slope and drainage analysis are to be conducted (favors raster). In many cases, the raster to vector conversion is done outside of the GIS in specialist conversion software, so that care can be taken to avoid the most common types of error, and so that the user can be brought in to resolve cases where the software is unable to solve a rasterization problem.

Increasingly, of course, many GIS systems allow the user to input and keep data in both raster and vector form. The GIS user should realize, however, that virtually all cross-structure retrieval and analysis requires one (or both) of the layers to change structure, and that this transformation often stamps itself irretrievably on the data's form, accuracy, and suitability for further use.

8.5 CHOOSING THE "BEST" GIS

The term "best" is extremely subjective where GIS is concerned. Some systems have extremely loyal followings who advocate their system over others. A "best" system implies that one solution is best for all problems, which is of course largely meaningless. The following subset of GIS systems, most available commercially, is intended to illustrate the breadth and depth of systems on the market today and some of the major and minor differences among these systems.

No endorsement is intended, and the list is provided to further the GIS "consumer's" education. Research has shown that these "big eight" packages account for the majority of those used in educational, and many professional, settings. In some cases, different GIS software packages are used in combination or along with other software for statistical analysis, graphical editing, or database management.

8.5.1 The Big Eight

ArcGIS. ArcGIS, the latest version of Arc/Info (Figure 8.9), is a long-lived, full-function GIS package that has been ported to the microcomputer, the workstation, and the mainframe. Arc/Info and ArcGIS are used to automate, manipulate, analyze, and display geographic data, and the software incorporates hundreds of sophisticated tools for map automation, data conversion, database management, map overlay and spatial analysis, interactive display and query, graphic editing, and address geocoding. The ArcInfo software includes a relational database interface for integration with commercial database management systems and a macro language called AML (ARC Macro Language) for developing customized applications. Since release 8, ArcGIS has instead used Visual Basic as its macro and programming language. ArcGIS uses a generic approach to geographic information systems that is not application specific, allowing the software to address virtually any geographic application. The software runs both on higher-end microcomputers and is available on several Unix workstations and for Windows NT. ArcGIS runs only on Windows NT.

ESRI is broadly accepted as a market leader in GIS, with many thousands of users in a variety of organizations worldwide. The software is used by federal, state, and local government organizations; and by businesses, utilities, and universities to address applications in planning, cartography, transportation, research, telecommunications, oil and gas, forestry, and many other disciplines. Release 8 of the program, in 1999, was a substantial modification of the program's user interface and functionality. Object-modeling capability and links to the Spatial Data Base Engine and other relational database management systems such as Oracle are included. With the latest versions of the software, the compatibility between ArcGIS and ArcView has been increased. The software uses the Windows COM component based software architecture, and is compatible with many other Window-based software tools.

ArcView. ArcView (Figure 8.10) is available for Windows, Macintosh, and a variety of Unix platforms. It is a desktop system for storing, modifying, querying, analyzing,

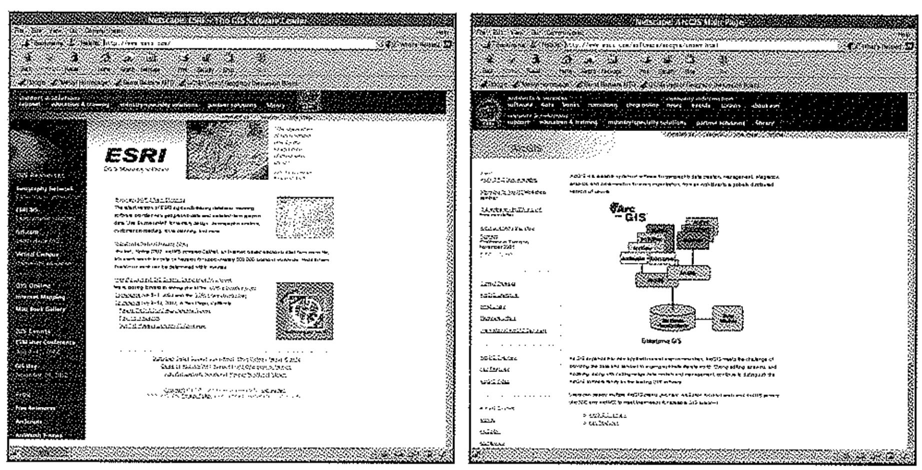


FIGURE 8.9: Arc/Info is a full-function GIS package. Left: The WWW home page for Arc/Info version 8 from the vendor, the Environmental Systems Research Institute at `http://www.esri.com`. (Used with permission, ESRI, Redlands, CA.)

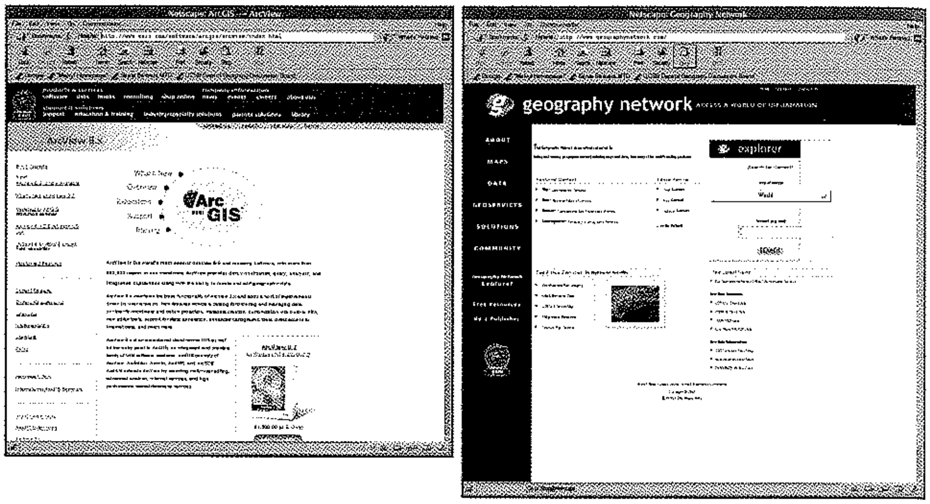


FIGURE 8.10: The ArcView GIS from ESRI Left: ArcView 8.1 product Web page at http://www.esri.com/base/products/arcview.html. Right: The ESRI geography network, source of many contributed shared data sets for use with ESRI software at http://www.geographynetwork.com.

and displaying information about geographic space. An intuitive graphical user interface includes data display and a viewing tool. Support for spatial and tabular queries, "hot links" to other desktop applications and data types, business graphics functions such as charting, bar and pie charts, and map symbolization, design, and layout capabilities are supported. Geo-coding and address matching are also possible. The Spatial Analyst tool kit makes working with raster data such as terrain and DEMs possible. Other extensions permit network analysis, allow Web activation of ArcView maps, and support advanced display features such as three-dimensional data visualization. ArcView GIS since version 8 has been more compatible with ArcGIS.

ArcView is also a product of ESRI, which makes ArcGIS. Compatibility exists between the two systems, with ArcView being more oriented toward map display than database management. Maps and data files are easily exchangeable between the formats used in the two systems, shape files, grid, images, and coverages. Outdated versions of ArcView have been placed into the public domain and are available over the Internet. A copy of the basic ArcView software is contained on the CD-ROM in the sleeve in the back of this book. More information is available at the ESRI Web site, cited in Figure 8.10.

Autodesk Map. Autodesk Map (Figure 8.11) is a GIS software suite built on the capabilities of the substantial AutoCAD software for automated drafting and design. Because this package is extensively used in planning, engineering, and architectural offices, many people can easily build upon their existing knowledge to enter the field of GIS. Autodesk Map uses AutoCAD 2002's drawing and plotting capabilities. Multiple data formats can be input, including those of AutoCAD (exchange format DXF and drawing format DWG) and also several other GIS packages. The software supports topology, query using Oracle and SQL, data management, and thematic mapping. The Autodesk Raster Design module supports grids and images and the Autodesk Onsite

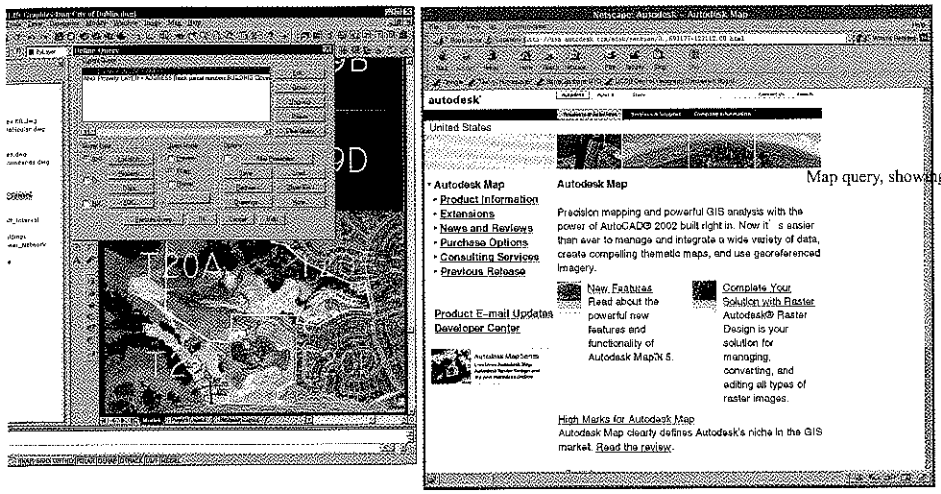


FIGURE 8.11: Autodesk Map from Autodesk. Home page at `http://www.autodesk.com` Left: Sample Autodesk Map query, showing use of Autodesk Map GIS functions within AutoCad 2002 CAD software.

module handles all of the standard GIS data operations. There are extensive tools for coordinate conversion and specification, rubber-sheeting, and map editing and digitizing. The software uses the C++ programming language as a development tool. Output control and plotting support are strong, relying on AutoCAD's capability.

GRASS. The U.S. Army Construction Engineering Research Laboratories (CERL) developed a public-domain software called the Geographic Resources Analysis Support System (GRASS). GRASS (Figure 8.12) is raster based, was the first Unix GIS software, and has been considerably enhanced by the addition of user contributions—for example, in hydrologic modeling. The Web site states that GRASS is an open source, free software GIS with raster, topological vector, image processing, and graphics production functionality that operates on various platforms through a graphical user interface and shell in X-Windows. The source code for the program is available under the GNU General Public License. The latest version, 5.0.0, the development version 5.1, and most prior versions are available free over the Internet. Many users run GRASS on PCs under the Linux version of Unix, although a Windows port is now complete. Since 1985, CERL has released upgrades and enhancements to GRASS and provided technical user support. However, CERL terminated GRASS-related work in the spring of 1996. Public domain user support has been very strong, and highly international.

Since 1996, the headquarters for GRASS support, research, and development has been at Baylor University, within the Department of Geology. The development currently under way is extensive, including releases of new manuals and documentation, and fosters continued research on GIS and visualization using GRASS at International conferences and through user support groups. The GRASS GIS uses a standardized command line input designed to resemble the Unix shell command language, but also uses a GUI under X-Windows. Unix compatibility allows users and programmers to create new applications and link GRASS to other software packages. Connections to the Unix shell and the C programing language allow simple extension and control.

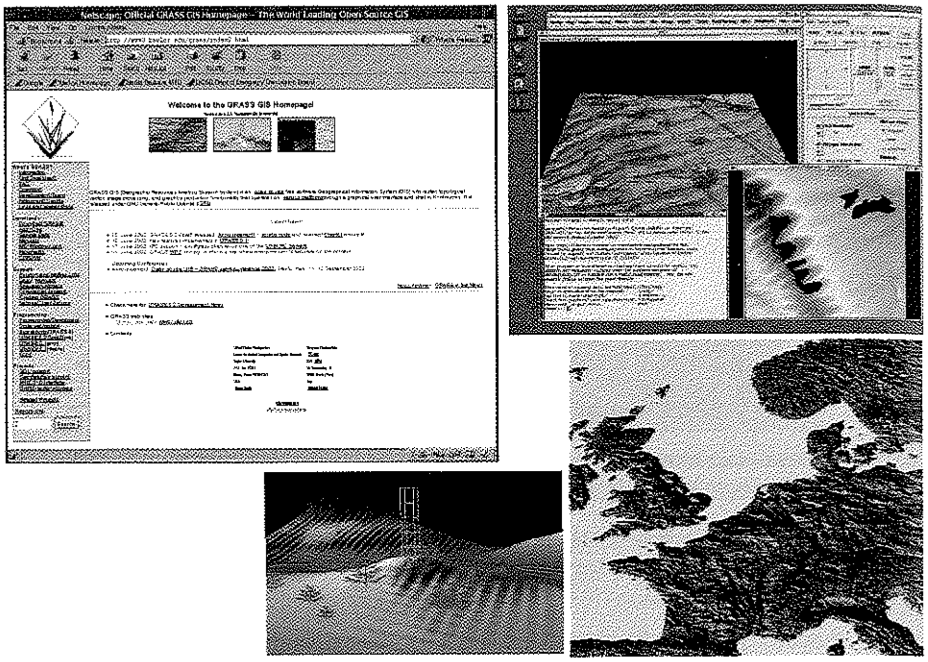
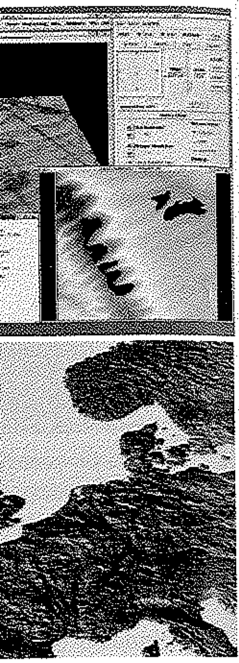


FIGURE 8.12: Left: The GRASS GIS Web page. (http://www3.baylor.edu/grass/index2.html) Right: The GRASS user interface: Bottom: Sample applications from the image galley.

IDRISI. The Idrisi (Figure 8.13) GIS software has been developed, distributed, and supported on a not-for-profit basis by the Idrisi Project, Clark University Graduate School of Geography. To date, there are many thousands of registered users of Idrisi software in almost every country in the world, making it the most broadly used raster GIS in the world. Idrisi is designed to be easy to use, yet provide professional-level GIS, image processing and spatial statistics analytical capability on both DOS- and Windows-based personal computers. It is intended to be affordable to all levels of users and to run on the most basic of common computer platforms. Expensive graphics cards or peripheral devices are not required to make use of the analytical power of the system, which is designed with an open architecture so that researchers can integrate their own modules.

Idrisi for Windows, first released in 1995, added a graphical user interface, flexible cartographic composition facilities, and an integrated database management system to the analytical tool kit. The more recent Idrisi32 is fully Windows and COM compliant and exploits object-oriented methods. Special routines for change and time-series analysis, spatial decision support, and uncertainty analysis and incorporation are included. A stand alone cartographic product, CartaLinx, allows topological editing and database development. Idrisi32 comes with a set of tutorial exercises and data that guide the new user through the concepts of GIS and image processing while also introducing the features of Idrisi. The tutorial exercises are appropriate for use either in self-training or in classroom settings.

MapInfo. MapInfo (Figure 8.14) was one of the first GIS programs to do desktop mapping. The vendor is MapInfo Corporation of Troy, New York. The software is well



aylor.edu/grass/index2.
image galley.

developed, distributed,
ark University Graduate
egistered users of Idrisi
broadly used raster GIS
ssional-level GIS, image
OS- and Windows-based
of users and to run on
ohics cards or peripheral
of the system, which is
grate their own modules.
al user interface, flexible
management system to
ws and COM compliant
ge and time-series anal-
orporation are included.
cal editing and database
data that guide the new
also introducing the fea-
ther in self-training or in

S programs to do desktop
ork. The software is well

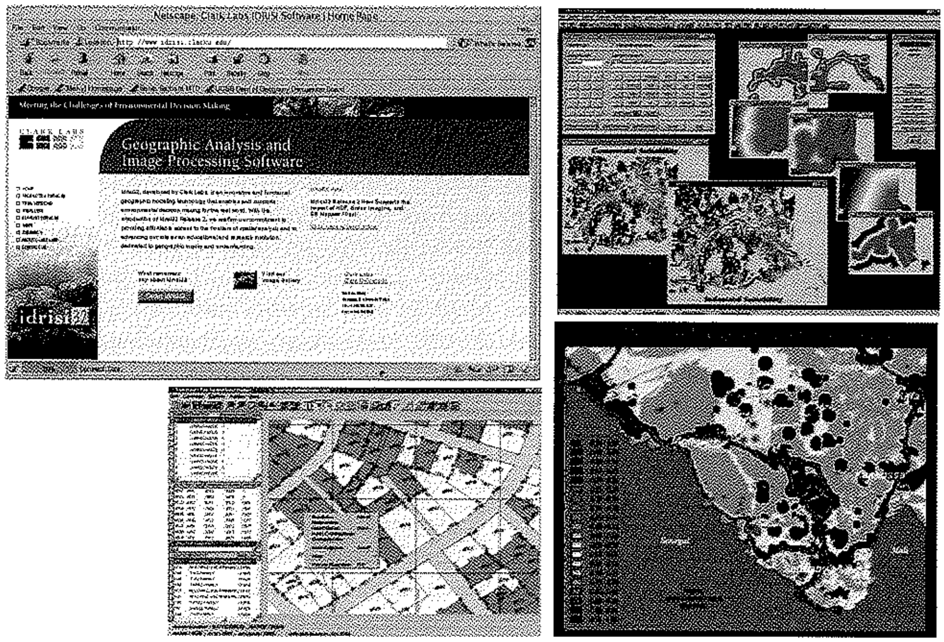


FIGURE 8.13: Left: The IDRISI project at `http://www.idrisi.clarku.edu`. Right side: Applications and examples of the Idrisi32 user interface taken from the Web site. (Courtesy of the IDRISI Project, Worcester, MA. Used with permission.)

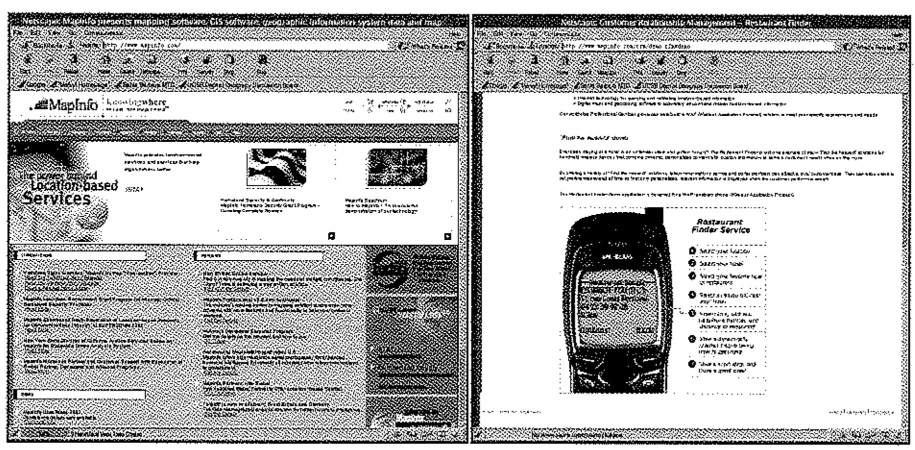


FIGURE 8.14: Left: The MapInfo home page at `http://www.mapinfo.com`. Right: MapInfo application, providing a mobile Internet search capability for restaurants using a cellular telephone. (Courtesy of MapInfo Corporation, Troy, NY. Used with permission.)

distributed and has many user groups and a broad variety of applications worldwide. The software runs under DOS, Windows, Macintosh, and on various Unix platforms. Although MapInfo's GIS retrieval and analysis functions are fewer than those of full-blown GIS packages, MapInfo includes a link to the Basic programming language via a language called MapBasic. This development environment permits the creation of customized "mapplications," extending MapInfo's built-in functionality and allowing use of a common graphical interface.

MapInfo has several GIS products aimed at different applications area, including MapInfo Professional, MapInfo MapX for programming GIS functionality, and specialist analytical modules such as MapXtreme for Web services, MapXSite for managing spatially enabled Web sites, and various database tools such as MapInfo Spatialware, Proviewer, and GIS Extension. MapInfo also supplies information products spanning geographic, economic, political, cultural, and industry application-specific content, each derived from leading worldwide sources to work the software. MapInfo also has an extensive training program, with classes at introductory and advanced levels for MapInfo and MapBasic.

Maptitude. Maptitude (Figure 8.15) is a GIS that works under the Windows operating system. The software is by Caliper Corporation, Newton, Massachusetts. Caliper has long been associated with the TransCAD and GIS-Plus GIS software packages. The latest version is Maptitude 4.5, which includes census data, a developer's toolkit and extended file support. The software comes with a considerable amount of geocoded and system-ready data on CD-ROM. The two CD-ROMs contain every street in the United States with the address information, state, county, zip codes, and census tracts as polygons with associated demographic data, and additional assorted U.S. and global data. Maptitude reads most standard PC file formats directly and can match each record against geographic data files using street address, zip code, and other features. Maptitude allows users to create and maintain geographic databases, analyze geographic relationships in data, and create highly professional map displays for presentations and reports.

Maptitude runs under Windows 3.1, Windows for Workgroups, Windows 95, or Windows NT, and with networks. The software uses the object linking and embedding of Windows, so that objects can be dragged and dropped into other applications.

GeoMedia. GeoMedia (Figure 8.16) is a widely distributed layer-based GIS with a tradition in computer-assisted design by the Intergraph Corporation of Huntsville, Alabama. The software runs on workstations, PCs, and under the Windows NT system. An extensive set of add-on modules allow users to configure GIS capability around their specific needs. The set of modules includes GeoMedia, GeoMedia Professional, Intelliwhere Ondemand (for mobile systems), GeoMedia Webmap, and GeoMedia WebMap Professional. There are extensions aimed at applications in land information, parcel management, public works, and transportation. The layered implementation permits efficient storage structures for the geometry and linkages to relational database records. Geographic elements are represented in the GIS as features. Features are grouped into the same categories as the maps on which they appear.

For the attribute data, GeoMedia incorporates use of the Oracle and SQL relational interface system, which facilitates client–server network communication to the relational DBMS so that multiple workstations communicate with the database server simultaneously. GeoMedia is fully integrated with Intergraph's traditional products, which

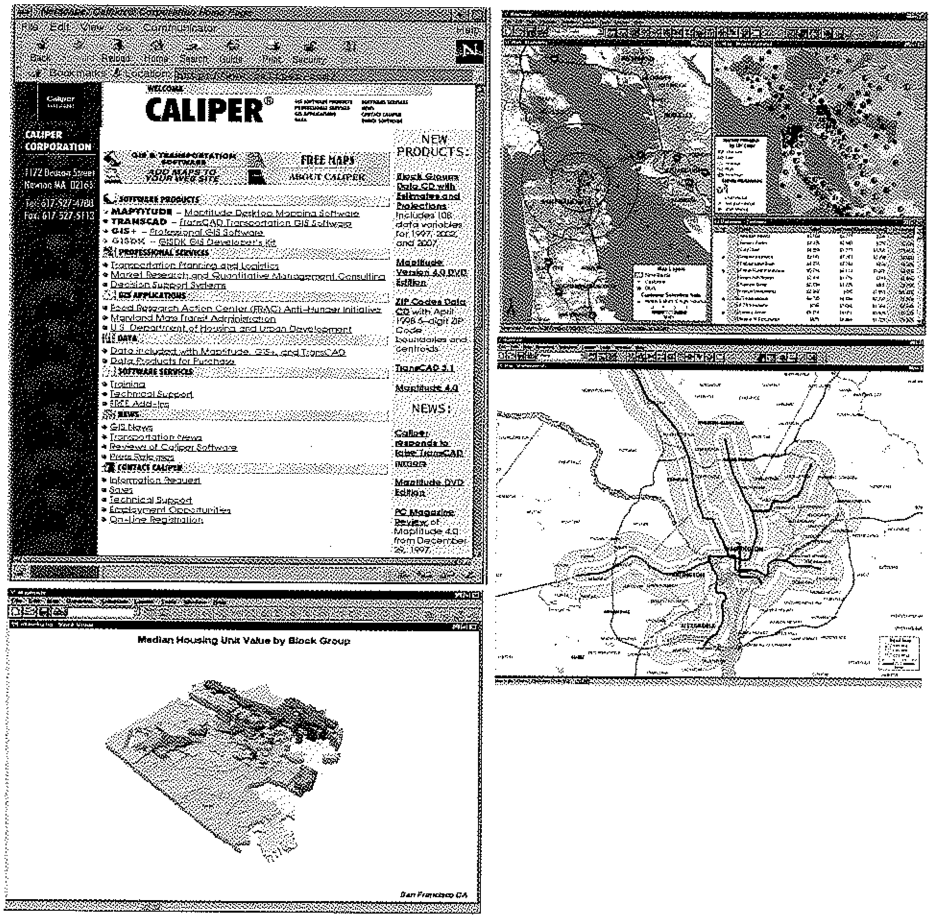


FIGURE 8.15: Upper left: The Caliper Corporation WWW home page at `http://www.caliper.com`. Right side: Sample applications and examples of the user interface. Bottom left: Example of a stepped statistical surface map generated by Maptitude. (Courtesy of Caliper Corp., Newton, MA. Used with Permission.)

include the MGE suite and tools for cartographic production. GeoMedia contains tools for building and maintaining topologically clean data without the processing and storage overhead of building and maintaining topology. In addition, it supports the open geodata interoperability specification and the spatial data transfer standard.

8.5.2 Selecting Software: Issues

Selecting the best GIS for use involves many other aspects than simply the technical capabilities of the software package. It could be argued that very little difference actually exists between GIS packages other than their user interfaces and their data structures. Conversely, many of the issues that determine how satisfied we are with the GIS we choose relate to how we acquire the software, how easily it installs itself on our computer, whether or not it is flexible enough to run on a given computer system, and how satisfied we are when the software is up and running.

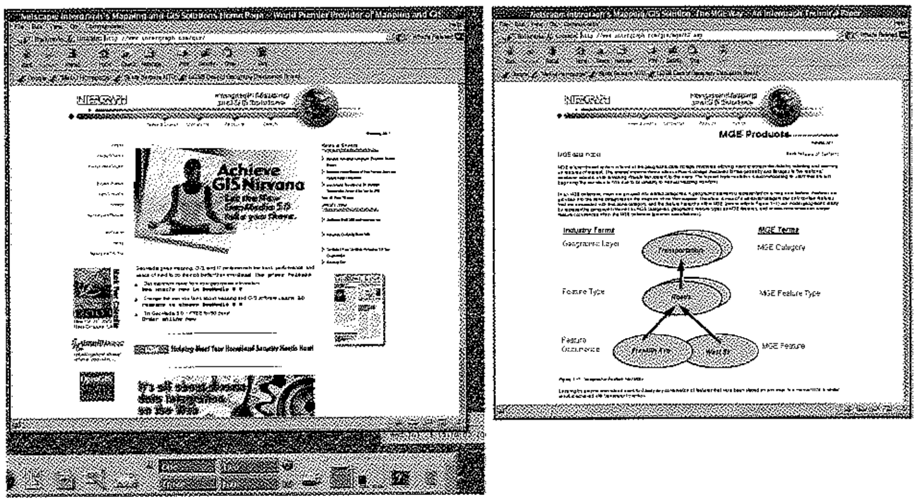


FIGURE 8.16: Left: The Intergraph home page at `http://www.intergraph.com/gis`. Right: Web page showing the basic data structure of Microstation MGE. (Courtesy of Intergraph Corp. Huntsville, AL. Used with permission.)

Obviously, cost is an important factor. Although the cost of basic GIS packages has fallen remarkably in recent years, cost can still be significant, especially when the hidden costs are taken into account. For example, GIS companies may charge not only a software purchase fee, but also include a maintenance fee, a fee for upgrades, a per call support cost, and sometimes other fees. Maintenance fees for workstation licenses, the sorts of licenses that would be used in a local area network configuration, can be a major proportion of the software cost. In addition, there is constant pressure to upgrade to new versions, usually by discontinuing support for older versions of the software. Especially if a large project is to be undertaken by the GIS user, this fact should be budgeted into the GIS software costs. Shareware and freeware, by contrast, may have less support infrastructure, but the software and update costs are zero.

Training is another important factor. Few GIS packages can be used by a novice right out of the box. The user may need help from a systems expert, may have special installation requirements, and may require the user to get some formal GIS training. Of course, this book can go a long way toward helping the user to understand GIS, but there is a great deal of straightforward technical information as well. Many GIS users take technical training from one of the GIS vendors or from other sources. These vary from one- or two-day workshops to entire college semester classes. They can also be rather expensive and time-consuming. Many GIS implementations, although well thought out and organized, fail for the lack of one or two people with the right technical expertise at the right time.

Once technical training ends, the real GIS use begins. At this stage, late on a Friday evening with a project deadline looming, the usual sole self-help mechanism is the GIS system manual. Again, these vary considerably in readability, comprehensiveness, and user-friendliness. Some are excellent, others poor. The user should ask to see documentation before making a major GIS purchase, as users will spend many hours poring over these pages. Best of all are online manuals, which can be searched, may be context-sensitive, have hypertext links, and will be available on a computer while the software is running in another window. This feature is worth extra expense, since it can speed the early learning and still serve as a reference later.

Regardless of the GIS's self-help capabilities, sooner or later almost all GIS users will eventually call a help line or interact in some way with the GIS vendor's technical support staff. In most cases this is done exclusively by telephone, but increasingly companies use fax, E-mail, and network conference groups as help facilities. Help lines can involve being placed on hold for long periods, or worse, waiting to be called back after leaving a phone number. E-mail is far better and gets around the time-zone problem of phone lines. When contacting a help line, a concise statement of the problem and a full set of information, usually including the serial number and date of purchase of the software, will greatly speed up your call. In general, using the reference manual or user guide until there is no other means of finding information is far preferable to calling a help line. Remember, if all else fails, *read the manual!*

Software maintenance can be another major consideration. For example, most software is updated by complete version upgrades, which require a new installation, or by "patches," a self-contained fix for a specific problem in the software. Maintenance is more of a consideration for large and networked systems, but every user needs to be concerned about too many large files and about how critical data are to be backed up in case of emergency. A GIS should also not be seen as a static entity, but rather one that will grow and evolve. A system that is big or powerful enough for a small prototype project today will probably not be able to deal with the follow-up project. Fortunately, as time passes the hardware becomes faster and faster, the disks get bigger and bigger, and the cost actually remains the same or falls. Conversely, the expertise required to install, maintain, and use the system is also important and should be planned for. GIS technicians typically get experienced enough to compete for better jobs very quickly. This should also be a part of the GIS cost plan.

Picking a GIS is obviously a complex and potentially confusing process. The most productive approach to the problem is to adopt the attitude of someone about to purchase a new car. First, the GIS user should assemble all the available details about the system requirements, the functional capabilities, the system constraints, and so on. The car buyer could, for example, determine a need for four doors, power steering, at least 14 cubic feet of luggage space, and front-wheel drive. Next, these should be matched against the systems available. Perhaps a trade-off is necessary between capabilities? Next comes the visit to the car dealership, followed by a test drive. Many demonstration versions of GIS packages are available to give a flavor of the system use before the purchase. Some demo versions can be downloaded free over the Internet or are given away at GIS conferences. One such demonstration version of ArcView is included with this book.

Finally, "You pays your money and you takes your choice." After the fact, however, the car will need to be maintained and perhaps repaired. One day it may be traded in for a new car. Every one of these issues should be considered. Although every vehicle will probably allow you to get home from the dealership, fundamental differences exist between a sports car and a Sports Utility Vehicle. Just so with GIS. To summarize: Before you choose, *research*, *select*, *test*, and *question*. Fortunately for the new GIS user, the early days of GIS failures are now over. Technically, today a GIS is much like a reliable automobile. Where and how you drive, however, is still entirely up to you!

8.6 STUDY GUIDE

8.6.1 Summary

CHAPTER 8: How to Pick a GIS

- GIS users need to be aware of different GIS software products during system selection and beyond.
- Informed choice is the best way to select the best GIS.
- GIS software has evolved very rapidly over its brief history.
- A historical GIS "snapshot" was the IGC survey conducted in 1979.
- In the 1979 survey, most GISs were sets of loosely linked FORTRAN programs performing spatial operations.
- Many early computer mapping programs had evolved GIS functionality by 1979.
- In the early 1980s, the spreadsheet was ported to the microcomputer, allowing "active" data.
- In the early 1980s, the relational DBMS evolved as the leading means for database management.
- Addition of a single integrated user interface and a degree of device independence led to the first true GISs.
- The second generation of GIS software used graphical user interfaces (GUIs) and the desktop/WIMP model.
- Unix workstations integrated GIS with the X-Windows GUI.
- As GUIs became part of the operating system, GISs began to use the operating system's GUI instead of their own.
- PCs integrated GIS with the variants of Windows and other OSs.
- GIS features are known as functional capabilities.
- Functional capabilities fall into the "critical six" categories.
- The critical six functional capabilities are data capture, storage, management, retrieval, analysis, and display.
- Some data capture functions are digitizing, scanning, mosaicing, editing, generalization, and topological cleaning.
- Storage functions are compression, metadata handling, control via macros or languages, and format support.
- Some data management functions are physical model support, the DBMS, address matching, masking, and cookie cutting.
- Some data retrieval functions are locating, selecting by attributes, buffering, map overlay, and map algebra.
- Some data analysis functions are interpolation, optimal path selection, geometric tests, and slope calculation.
- Some data display functions are desktop mapping, interactive modification of cartographic elements, and graphic file export.
- Many GIS functional capabilities are by-products of their particular data structure.
- Raster systems work best in forestry, photogrammetry, remote sensing, terrain analysis, and hydrology.
- Vector systems work best for land parcels, census data, precise positional data, and networks.

- **Eight GIS systems form the bulk of operational GIS in professional and educational environments.**
- **There are some significant differences among these "big eight" systems.**
- **A variety of issues should be considered in system selection:**
  - **cost**
  - **upgrades**
  - **network configuration support**
  - **training needs**
  - **ease of installation**
  - **maintenance**
  - **documentation and manuals**
  - **help-line and vendor support**
  - **means of making patches**
  - **workforce**
- **Selecting a GIS can be a complex and confusing process.**
- **The intelligent GIS consumer should research, select, test, and question systems before purchase.**

8.6.2 Study Questions

The Evolution of GIS Software

Make a timeline from about 1960 to today. Place on the timeline each of the packages mentioned in this chapter. How does the sequence of software packages relate to the history of GIS discussed in Chapter 1?

GIS and Operating Systems

Make a list of all the operating systems, mainframes, workstations, and microcomputers that can run the "big eight" GIS packages. Which are mentioned most frequently? Why?

GIS Software Capabilities

Make a word list of key functional capabilities structured under the headings of the "critical six." Score the functional capabilities by how essential they are for a GIS to qualify as a "true" GIS. Match the capabilities of one particular GIS against the list.

GIS Software and Data Structures

Review the Chapter 4 coverage of the different data structures for GIS. Classify each of the "big eight" by which data structure they support and whether or not they support data structure conversions. List some operational reasons why you might need to convert between data structures.

Choosing the Best GIS

Go through the "People in GIS" sections in this book and tally the mentions of specific GIS software packages. How does your list match up with the "big eight"?

Make a table of the "critical six" as columns and the "big eight" as rows. Fill in the table entries with observations from the text on the "big eight." Invent a scoring system and rank the "big eight" for their suitability in each of the applications areas listed in Section 8.4.

8.7 EXERCISES

1. *If you have access to more than one GIS, establish a common data set, such as a TIGER file or a DLG, read the file into the GIS system, and perform a simple retrieval or analysis operation such as a buffer or overlay. Take careful note of how long each step took, how many steps were necessary, and how useful the manuals and help systems were in troubleshooting. Place the two output maps together at the same size and scale. Are they identical? What might be the factors contributing to the differences?*
2. *Examine the manuals for two different GIS packages. Read the same section—for example, the section on digitizing lines—in each manual. Which is the better explanation? Why? Make a list of the features that you consider desirable in GIS documentation.*
3. *If you have the ability, install another operating system on your workstation or microcomputer, such as Linux and Windows. Alternatively, find two computers with different operating systems already installed. Do the same task—say, enter 50 numbers into a spreadsheet file—in each of the two operating systems. Time each process and make a chart showing how much total time you spent on each task. How much did the operating system help or hinder the task? How much system help was available in each system?*
4. *Using the Internet, the mail, or any other means available, make a comparative price list of the "big eight" and as many other GIS packages as you can find. Using the same functions checklist from the study questions, compute a "features per dollar" number for each GIS. Which is the best? Which is the worst? Why?*
5. *Follow the network conference group comp.infosystems.gis for one week and keep a tally of good and bad comments about the GIS systems that are discussed there. Would this be a good way to choose a GIS? Why or why not?*

8.8 REFERENCES

Brassel, K. E. (1977) "A survey of cartographic display software," *International Yearbook of Cartography*, vol. 17, pp. 60–76.

Day, D. L. (1981) "Geographic information systems: all that glitters is not gold." *Proceedings, Autocarto IV*, vol. 1, pp. 541–545.

Marble, D. F. (ed.) (1980) *Computer Software for Spatial Data Handling*. Ottawa: International Geographical Union, Commission on Geographical Data Sensing and Processing.

Tomlin, D. (1990) *Geographic Information Systems and Cartographic Modelling*. Upper Saddle River, NJ: Prentice Hall.

Tomlinson, R. F. and Boyle, A. R. (1981) "The state of development of systems for handling natural resources inventory data," *Cartographica*, vol. 18, no. 4, pp. 65–95.

8.9 KEY TERMS AND DEFINITIONS

active data: Data that can be reconfigured and recomputed in place. Spreadsheet term for data for attributes or records created by formulas within a spreadsheet.

address matching: Address matching means using a street address such as *123 Main Street* in conjunction with a digital map to place the street address onto the map in a known location. Address matching a mailing list, for example, would convert the mailing list to a map and allow the mapping of characteristics of the places on the list.

affine transformation: Any set of translation, rotation, and scaling operations in the two spatial directions of the plane. Affine transformations allow maps with different scales, orientations, and origins to be coregistered.

Autodesk Map: A GIS software package. See Section 8.5.

batch: Submission of a set of commands to the computer from a file rather than directly from the user as an interactive exchange.

big eight: The eight most popular GIS packages, established by the numbers of users, particularly among people getting started with GIS, at any given time.

buffer: A zone around a point, line, or area feature that is assumed to be spatially related to the feature.

CALFORM: An early computer mapping package for thematic mapping.

CAM (computer-assisted mapping): A map projection and outline plotting program for mainframe computers dating from the 1960s.

CGIS (Canadian Geographic Information System): An early national land inventory system in Canada that evolved into a full GIS.

clump: To aggregate spatially; to join features with similar characteristics into a single feature.

compression: Any technique that reduces the physical file size of data in a spatial or other data format.

cookie-cut: A spatial operation to exclude area outside a specific zone of interest. For example, a state outline map can be used to cut out pixels from a satellite image.

critical six: The GIS functional capabilities included in Dueker's GIS definition: map input, storage, management, retrieval, analysis, and display.

data exchange format: The specific physical data format in which exchange of data between similar GIS packages takes place.

data structure: The logical and physical means by which a map feature or an attribute is digitally encoded.

DBMS (database management system): Part of a GIS; the set of tools that allow the manipulation and use of files containing attribute data.

desktop mapping: The ability to generate easily a variety of map types, symbolization methods, and displays by manipulating the cartographic elements directly.

desktop metaphor: For a GUI, the physical analogy for the elements with which the user will interact. Many computer GUIs use the desktop as a metaphor, with the elements of a calendar, clock, files and file cabinets, and so on.

device independence: The ability of software to run with little difference from a user's perspective on any computer or on any specialized device, such as a printer or plotter.

dissolve: Eliminating a boundary formed by the edge or boundary of a feature that becomes unnecessary after data have been captured: for example, the edges of sheet maps.

Dueker's definition of GIS: "A special case of information systems where the database consists of observations on spatially distributed features, activities or events, which are definable in space as points, lines, or areas. A geographic information system manipulates data about these points, lines, and areas to retrieve data for ad hoc queries and analyses."

DXF: Autocad's digital file exchange format, a vector-mode industry standard format for graphic file exchange.

dynamic segmentation: GIS function that breaks a line into points at locations that have significance, and that can have their own attributes. For example, the line representing a highway can have a new node added every mile as a mile marker that can hold attributes about the traffic flow at that place.

edge matching: The GIS or digital map equivalent of matching paper maps along their edges. Features that continue over the edge must be "zipped" together and the edge dissolved. To edge-match, maps must be on the same projection, datum, ellipsoid, and scale and show features captured at the same equivalent scale.

entity by entity: Any data structure that specifies features one at a time, rather than as an entire layer.

FORTRAN: An early computer programming language, initially for converting mathematical formulas into computer instructions.

functional capability: One of the distinctive processes that a GIS is able to perform as a separate operation or as part of another operation.

functional definition: Definition of a system by what it does rather than what it is.

fuzzy tolerance: Linear distance within which points should be snapped together.

generalization: The process of moving from one map scale to a smaller (less detailed) scale, changing the form of features by simplification, and so on.

geometric test: A test to establish the spatial relationship between features. For example, a point feature can be given a point-in-polygon test to find if it is "contained" by an area.

GNU: Free Software Foundation organization that distributes software over the Internet.

GRASS: A GIS software package (see Section 8.5).

GUI (Graphical User Interface): The set of visual and mechanical tools through which a user interacts with a computer, usually consisting of windows, menus, icons, and pointers.

help line: A telephone service available to software users for verbal help from an expert.

import: The capability of a GIS to bring data in an external file and in a nonnative format for use within the GIS.

installation: The step necessary between delivery of GIS software and its first use, consisting of copying and decompressing files, data, registering licenses, and so on.

integrated software: Software that works together as part of a common user interface rather than software that consists of separate programs to be used in sequence.

local area network: An arrangement of computers into a cluster, with network linkages between computers but no external link. Usually, this allows sharing data and software licenses, or the use of a file server.

macro: A command language interface allowing a "program" to be written, edited, and then submitted to the GIS user interface.

map algebra: Tomlin's terminology for the arithmetic of map combination for coregistered layers with rasters of identical size and resolution.

map overlay: Placing multiple thematic maps in precise registration, with the same scale, projections, and extent, so that a compound view is possible.

Maptitude: A GIS software package (see Section 8.5).

mask: A map layer intended to eliminate or exclude areas not needed for mapping and analysis.

metadata: Data about data. Index-type information pertaining to the entire data set rather than the objects within the data set. Metadata usually includes the date, source, map projection, scale, resolution, accuracy, and reliability of the information, as well as data about the format and structure of the data set.

mosaic: The GIS or digital map equivalent of matching paper maps along their edges. Features that continue over the edge must be "zipped" together and the edge dissolved. To edge-match, maps must be on the same projection, datum, ellipsoid, and scale, and show features captured at the same equivalent scale. See also **edge matching**.

Motif: A graphical user interface standard that is common on Unix workstations.

multitask: The ability of a computer's operating system or a GIS to handle more than one process at once; for example, editing and running a command sequence while extracting data from the database and displaying a map.

node snap: Instructing the GIS software to make multiple nodes or points in a single node so that the features connected to the nodes match precisely, say at a boundary.

online manual: A digital version of a computer application manual available for searching and examination as required.

patch: A fix to a program or data set involving a sequence of data that are to be overwritten onto an older version.

proprietary format: A data format whose specification is a copyrighted property rather than public knowledge.

raster: A data structure for maps based on grid cells.

relational DBMS: A database management system based on the relational data model.

renumbering: Use of the DBMS to change the ordering or ranges of attributes. Also, especially in raster GISs, to change the numbers within grid cells into categories.

rubber sheeting: A statistical distortion of two map layers so that spatial coregistration is accomplished, usually at a set of common points.

sift: To eliminate features that are smaller than a minimum feature size.

spatial data transfer standard (SDTS): The formal standard specifying the organization and mechanism for the transfer of GIS data between dissimilar computer systems. Adopted as FIPS 173 in 1992, SDTS specifies terminology, feature types, and accuracy specifications as well as a formal file transfer method for any generic geographic data. Subsets for the standard for specific types of data—vector and raster, for example—are called profiles.

spreadsheet: A computer program that allows the user to enter numbers and text into a table with rows and columns, and then maintain and manipulate those numbers using the table structure.

SURFACE II: An early computer mapping package from the Kansas Geological Survey.

SYMAP: An early multipurpose computer mapping package.

topologically clean: The status of a digital vector map when all arcs that should be connected are connected at nodes with identical coordinates and the polygons formed by connected arcs have no duplicate, disconnected, or missing arcs.

Unix: A computer operating system that has been made workable on virtually every possible computer and has become the operating system of choice for workstations and science and engineering applications.

upward compatibility: The ability of software to move on to a new version with complete support for the data, scripts, functions, and so on, of earlier versions.

user interface: The physical means of communication between a person and a software program or operating system. At its most basic, this is the exchange of typed statements in English or a programlike set of commands.

vector: A map data structure using the point or node and the connecting segment as the basic building block for representing geographic features.

version: An update of software. Complete rewrites are usually assigned entirely new version numbers (e.g., Version 3), while fixes and minor improvements are given decimal increments (e.g., Version 3.1).

VisiCalc: A spreadsheet package for first-generation microcomputers. Supported data tables in flat files.

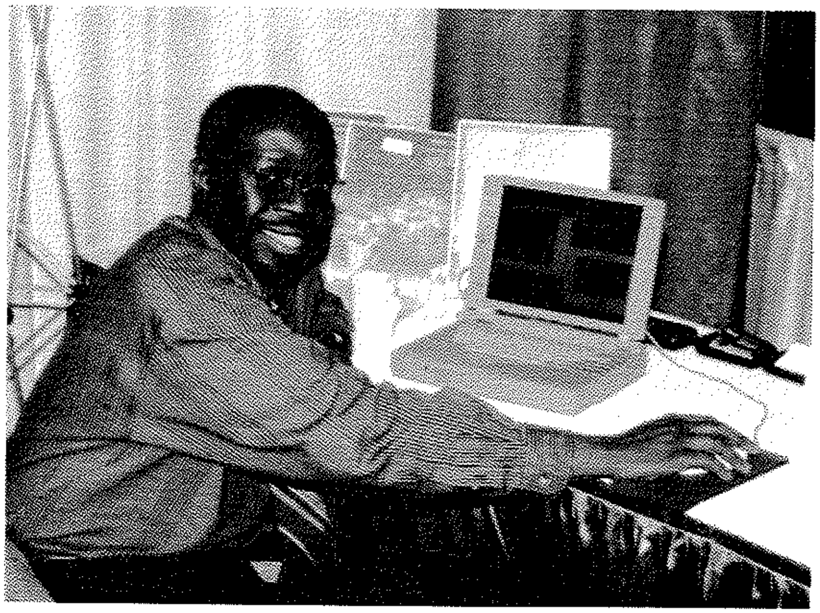
warping: See **rubber sheeting.**

WIMP: A GUI term reflecting the primary user interface tools available: windows, icons, menus, and pointers.

X-Windows: A public-domain GUI built on the Unix operating system and computer graphics capabilities, written and supported by the Massachusetts Institute of Technology and the basis of most workstation shareware on the Internet.

zip: See **mosaic.**

PEOPLE IN GIS



Assaf Anyamba Research Associate, The IDRISI Project

Assaf Anyamba is a research associate at the Clark University Laboratories for Computer Cartographic Technology and Geographical Analysis, the IDRISI Project, based in Worcester, Massachusetts. His elementary, secondary, high school, and undergraduate education was undertaken in Kenya under the British educational system. Assaf has a B.A. with a double major in geography and economics from Kenyatta University, Nairobi, and an M.A. in geography from Ohio University. He is a Ph.D. student and a

ı person and a software
nge of typed statements

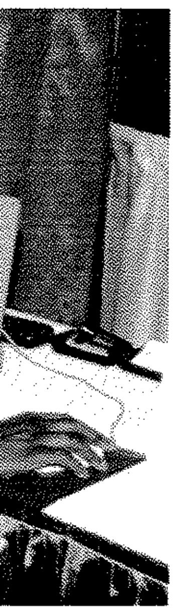
ınecting segment as the

assigned entirely new
nprovements are given

puters. Supported data

ıilable: windows, icons,

system and computer
tts Institute of Technol-
et.



**Laboratories for Com-
IDRISI Project, based
chool, and undergrad-
ıcational system. Assaf
n Kenyatta University,
a Ph.D. student and a**

NASA Global Change Research Fellow at Clark, where he is working on a dissertation studying El Niño's impacts on Africa.

KC: Assaf, how did your GIS career begin?

AA: During my undergraduate years I had the opportunity to participate in the Koobi Fora Harvard University Field School Program in the Rift Valley region of Kenya. Later I spent two months at the Regional Center for Services in Surveying Mapping and Remote Sensing in Nairobi, annotating the Rift Valley Landsat Mosaic and worked as a research assistant on the Kenya Rift International Seismic Project with a team of scientists from Karlsruhe University Institute of Geophysics. The major aim of the project was to gain a better understanding of the deep structure of the rift valley system using digital geophysical techniques. This was my first introduction to digital "things." I went on to Ohio University, where I got my Master's degree in geography. My thesis was on the comparison of ecological variables with coarse resolution satellite data for ecological mapping over Kenya. While there, I worked one summer on a project involving electoral districting, my first experience of project-oriented GIS. I spent a summer at the NASA/Goddard Space Flight Center through a Universities Space Research Association graduate summer internship program working in the Global Inventory Monitoring and Modeling Systems (GIMMS) Lab. At Clark University I am pursuing a Ph.D., working on reconstructing the El Niño/southern oscillation from coarse resolution satellite time-series data for Africa. I have a NASA Graduate Fellowship in Global Change Research.

KC: When did you first hear about GIS?

AA: In 1988 from a professor in Kenya as "automated cartography" in a theoretical sense (we had no practical exposure) and from a UNEP brochure in 1989. I thought of GIS as "spatial database" organizing land resources information (land cover, land use, population, drainage, etc.).

KC: And how would you define GIS today?

AA: As computerized systems for input, archiving, and manipulating different forms of geographic data and for output of derivative products from the data to assist in providing answers to environmental questions and to highlight specific environmental problems or resources.

KC: What is your role at IDRISI?

AA: Primarily research. I am one in a group of project members that form the Change and Time Series Group. We are concerned with development and use of time-series analysis techniques to understand spatiotemporal change in environmental data. We want to apply these techniques to drought and famine early warning, food security issues, and climate variability. I also help with software testing, with the WWW site for the IDRISI Project, and in training.

KC: What do you see as developments in GIS that have made it a practical technology for use throughout the world, especially in developing nations?

AA: Perhaps the most revolutionary thing was "porting" the GIS engine to the PC platform. PCs are cheaply available worldwide and easy to use. They have made it possible for most developing countries to undertake GIS projects.

KC: What is important to someone just getting started in GIS?

AA: Education, education, education! There needs to be a revised curriculum at the university level in GIS, perhaps national standards. Most schools are training GIS software specialists rather than geographic information scientists. This may not be wrong but it does affect GIS implementation. Software specialists can be narrow-minded and lose track of the broad scientific view. Training of GIS specialists needs to be stressed and monitored, perhaps by a GIS standards consortium. GIS education should cover basic geodesy and computer science; and ecology, climatology, biology, demography, and so on, should have a GIS component attached to them.

KC: Thanks Assaf. (Used with permission.)