reports for most of these results.

NTIS also manages the Federal Computer Products Center which provides access to software, data files, and databases produced by Federal agencies.

To order reports and computer products, call: (703) 487-4650.

Line simplifications and digital cartographic databases. H. Walker. Lawrence Livermore National Lab, CA, November 1989, pp. 227, UCRL-LR-103168. Portions of this document are illegible in microfiche products. Original copy available until stock is exhausted. National Technical Information Service, 5285 Port Royal Rd., Springfield, VA 22161. DE90010194/WNR, price code: PC A11/MF A01.

This thesis examines one component of the cartographic generalization process, line simplification, and its implementation with computer technology. The effectiveness of line simplification techniques in producing useful maps from digital map data is studied. Many techniques have been proposed to simplify lines; several of these have been implemented here. The approaches have been compared on their ability to simplify lines from two cartographic databases over both modest and extreme changes in scale. The comparisons are based on the simplification of individual lines, on the production of complete maps, and on mathematical measures of performance. One promising technique is enhanced to overcome certain limitations which are apparent during extreme scale change. The effect of map projections on the performance of such techniques is considered. While much additional work is needed, this study indicates that line simplification techniques can extend the range of scales over which certain cartographic databases can be utilized. 124 refs.

Computer Program for Converting Rectangular Coordinates to Latitude-Longitude Coordinates. A. T. Rutledge. Geological Survey, Tallahassee, FL, Water Resources Division 1989, 21 pp. USGS/WRI-89-4070. Also available from Superintendent of Documents. PB90-231796/WNR; price code: PC A03/MF A01.

A Fortran-77 computer program was developed for converting the coordinates of any rectangular grid on a map to coordinates on a grid that is parallel to lines of equal latitude and longitude. Using the program in conjunction with ground-water flow models, the user can extract data and results from models with varying grid orientations and place these data into a grid structure that is oriented parallel to lines of equal latitude and longitude. All cells in the rectangular grid must have equal dimensions, and all cells in the latitude-longitude grid must measure 1 minute by 1 minute. The program is applicable if the map used shows lines of equal latitude as arcs and lines of equal longitude as straight lines and assumes that the Earth's surface can be approximated as a sphere.

Guidelines for Running GRASS Benchmarks. M. O. Johnson. Construction Engineering Research Lab (Army), Champaign, IL, February 1989, 11 pp. CERL-N-89/23. AD-A221 332/0/WNR; price code: PC A03/MF A01.

The Geographic Resources Analysis Support System (GRASS) is a geographic information and image processing system originally designed to serve land managers and environment planners at Army installations. GRASS is public domain software distributed by several public and private organizations. Consequently, there are many different hardware configurations running GRASS. This guide documents current procedures used to conduct system performance tests (benchmarks), to provide users considering the acquisition of the system a reliable means to compare the many systems that run GRASS. Relevant data that accompany benchmark results are: machine specifications; system environment; GRASS program environment; GRASS graphics environment; operation descriptions; data description; and benchmark execution notes.

cart lab bulletin board

This forum is offered to encourage communication among practitioners at a time of rapid technological transition. Questions, comments, and announcements are invited.

INTRODUCTION TO MACINTOSH GRAPHICS FILE FORMATS

David DiBiase
Deasy GeoGraphics , Penn State

Maps and other information graphics can be produced with microcomputers, but no one microcomputer-based software package is adequate for all cartographic production tasks. No matter whether Apple Macintosh or IBM-compatible PC hardware platforms are used, microcomputer cartographers often must pass a digital map file from one special-purpose software package to another. Successful file transfers result when both the exporting and importing packages support the same file format specification, or when a file exported by one program is reformatted to a specification compatible with the importing program. Familiarity with characteristics of various

digital file formats is therefore useful.

This note presents descriptions of several file formats common to graphics and page layout software for the Apple Macintosh. It concludes with observations about the status of PostScript, a page description language that facilitates graphics data transfer between software packages as well as control of high-resolution output devices. The descriptions are based on articles in trade journals, technical manuals, and discussions with Macintosh graphics users and developers, as well as on three years' personal experience creating graphics with Macintosh software. The discussion deals with just the few most common formats. Many programs offer their own proprietary options. Further, there is no absolute standardization of formats; users will encounter numerous variations between formats implemented by different applications.

Two types of formats

Graphics data formats for Macintosh graphics programs are of two types: *grid-based* (raster) and *object-based* (vector). Grid-based formats (such as MacPaint and TIFF) store image data as matrices of numerical values which correspond to black and white, gray scale or color pixels, depending on the range of values assigned to each grid cell. Object-based formats (PICT, PostScript and EPS) store lists of instructions by which line segments are plotted, but can also incorporate raster image data. Both types can store high-resolution, near-continuous tone gray scale and color image data.

GRID-BASED FORMATS

MacPaint

The MacPaint format was developed for the original Macintosh graphics program. It is based on a fixed point-system grid of 72 dots per inch (dpi). A single binary value is assigned to each grid cell, so that MacPaint files can only store monochrome imagery. Gray values are roughly approximated by dithered patterns; lines and typography are caricatured by jagged mosaics of black and white pixels. The MacPaint format is supported by most paint-type programs for the Macintosh, and can be generated by most desktop scanners. Its fixed, low resolution makes it inappropriate for professional graphics applications, except perhaps for the temporary storage of scanned drawing templates.

TIFF

The 'Tag Image File Format' was developed by Aldus Corp. specifically for storing gray scale image data captured with digital scanners and making it compatible with page layout programs like *PageMaker*. Unlike the fixed MacPaint format, TIFF allows for variation in file structure by including a directory of 'tag' headers that describe image data contained elsewhere in the file. The format imposes virtually no limits on grid resolution or the range of grays or colors that can be stored. Practical limits are imposed on the resolution of TIFF images by scanning hardware, graphics boards, and the amount of random access memory (RAM) installed in the computer. Near-photographic image quality can be preserved in the TIFF format, but at an enormous storage cost. For example, an 8.5" by 11" 8-bit gray scale image scanned at 600 dpi would require more than 33 Mb of memory — too much to be manipulated on a Macintosh, even after its virtual RAM ceiling is raised with the forthcoming introduction of Apple's System 7. Fortunately, images are often smaller than page size, and the TIFF format supports various data compression schemes implemented in widely distributed utility programs like *StuffIt*. TIFF files can be generated and manipulated in paint-type packages such as *Studio/8*, *PixelPaint* and *DeskPaint*, image retouching packages like *PhotoShop*, *ImageStudio* and *Digital Darkroom*, and packages that integrate bitmap and object editing tools, such as *Canvas*, *MacDraw II* and *SuperPaint*. A version of the format exists in the DOS world (TIF), and utility programs are available to translate the DOS form to the Mac (such as Hewlett Packard's *TIF to TIFF*). TIFF is particularly useful where shading effects are required; *ImageStudio*, for example, can be used to create high-resolution shaded relief. TIFF files are not well suited for iterative revisions of linework and typography, however, since these elements (like all others) are stored as grid patterns and cannot be addressed as discrete objects.

OBJECT-BASED FORMATS

PICT

Graphics data are stored as a list of 'QuickDraw' drawing instructions in PICT files. QuickDraw comes as part of every Macintosh's firmware, providing a set of procedures by which graphics and text are rendered on the Mac's monitor. PICT files contain drawing instructions for several types of two-dimensional objects, including vector drawings, raster images, and fonts. The resolution of an image described by a PICT file is limited only by application software and the display device. More problematic is the limited (1/72") precision of coordinate definitions in the PICT format, which can frustrate attempts to import detailed CAD drawings into PICT-based drawing programs. The earliest version of the PICT format supported only 1-bit monochrome images, but latest version (sometimes called "PICT2") supports up to 24-bit color.

Some version of the PICT format

is supported by virtually every graphics package for the Mac. PICT graphics can be easily passed among Macintosh-based mapping, statistical graphics and drawing programs via the clipboard, a portion of the Mac's main memory reserved for data transfer. The format supports "picture comments," statements which describe object groupings. Software packages that take full advantage of picture comments can generate PICT files in which objects can be "ungrouped" and graphically manipulated in other packages. This feature enables Mac users to cut and paste maps generated (for instance) in Strategic Mapping's *MapMaker* into an Aldus *FreeHand* document, preserving enumeration units as discrete objects whose graphic attributes can be modified. Not all packages take full advantage of picture comments, however. Graphics generated in some spreadsheets and statistical graphics programs cannot be ungrouped and customized in drawing packages.

Users who rely solely on Macintosh-based software find the PICT format satisfactory for many, but not all, applications. A few important Macintosh programs (Adobe's *Illustrator 3*, for example) do not support the PICT format. Those who need to integrate more than one computing system will find file format conversion utilities that translate PICT files (Kandu Software's *CAD Mover* [which runs on Macs] and Inset Systems' *HiJaak* [which runs on PCs], for example) very useful. Since PICT files are coded in machine language—which fewer people know how to edit than ASCII text—they are not easily transformed by the end user.

PostScript and EPS

Along with microcomputers and page layout software, laser printers constitute the technological base of the $1 billion "desktop publishing" industry. More than half of the laser printers on the market are PostScript-compatible (Anderson 1990). Laser printers (including laserwriters and imagesetters) interpret graphics and text files sent from software applications as PostScript programs in ASCII text form into bitmaps that are imaged with toner on plain paper or direct to photographic films and papers. PostScript is a high-level programming language developed by Adobe Systems Inc. specifically for describing the appearance of graphics and text on printed pages. Like PICT files, PostScript documents can describe two-dimensional vector or raster objects whose resolution is limited only by the display device. Unlike the PICT format, however, PostScript does not limit the precision of coordinate positions to 1/72". PostScript also exceeds the PICT format in supporting process color separations.

Adobe has documented and released to the public domain most of the specifications of the PostScript language, but guarded details of its PostScript interpreter technology, which it licenses to laser printer manufacturers. The company has also produced a proprietary library of more than 550 fonts coded as PostScript objects with 'hints' that guide the necessary non-linear scaling of letterform proportions at extreme sizes. PostScript has become the *de facto* standard imaging model for printed text and graphics in the publishing industry. Nearly every text- or graphics-generating software package for microcomputers offers driver software that generates PostScript. The most powerful illustration programs for thematic cartography and information graphic design on the Macintosh (such as *Illustrator* and *FreeHand)* are those based on PostScript, because these are most compatible with high-resolution PostScript laser printers.

PostScript is an extensible language that does not require driver developers to conform to a standardized program format. The lack of a standardized format explains why PostScript files written to disk from mapping and statistical graphics programs usually cannot be opened and graphically edited in illustration programs. Fortunately, since PostScript files are coded in ASCII text, they can be reformatted by end users without expertise in machine language programming. Utility programs that convert PostScript files to the Adobe Illustrator file format show promise for linking mapping and illustration programs so that cartographers can take advantage of the specialized capabilities of both (DiBiase, forthcoming).

The "Encapsulated PostScript Format" (EPS) consists of ordinary PostScript ASCII code with a corresponding 72 dpi PICT-format screen image imbedded. The format was developed to ease the integration of graphics with text in page layout programs like Aldus' *PageMaker*. When a user places an EPS-format graphic into a page layout document, a bitmapped screen representation is added to the page layout file along with a link to the original EPS file. When the document is directed to a printer, the page layout program's PostScript driver ignores the screen image and sends the PostScript drawing instructions from the EPS file to the output device. This redundancy of image information has serious storage implications, and reflects the unfortunate duality in screen and printer imaging models adopted by both Macintosh and DOS-based microcomputer systems. Workstations like the NeXT and Sun avoid this problem by employing PostScript both as a display and printing model, though the relatively slow response rate of display PostScript

causes some dissatisfaction.

A powerful feature of the EPS format is that it allows the user to resize an image without affecting the resolution on the printed page. Unlike an ordinary PostScript document, however, the elements of an EPS file cannot readily be manipulated in a drawing or painting package, since two sets of image data (using different image models) would have to be modified simultaneously.

Implications of the "font wars"

At the Seybold Computer Publishing Conference held in September, 1989, Microsoft President Bill Gates announced an agreement whereby Microsoft would develop a clone version of PostScript's interpreter technology in return for the right to use Apple's newly developed "TrueType" scalable outline font technology in its OS/2 Presentation Manager and Windows user interfaces. Most accounts in the trade press have since focused on Macintosh users' concern for the future compatibility of their expensive collections of Adobe type 1 fonts (hence "font wars"). A more important issue seems to be the adverse effect this deal is likely to have on the prospect of a standard imaging model for desktop publishing and information graphics.

Adobe Systems has reaped enormous profits from licensing fees for its PostScript interpreter technology and sales of its proprietary PostScript fonts. The desktop publishing industry's reliance on these products, coupled with Adobe's unyielding defense of its proprietary rights, led to the perception that the industry was contending with a PostScript "cartel." The Microsoft/Apple agreement (along with increasingly successful clone PostScript interpreters like Custom Application's *Freedom of Press)* has broken the cartel, threatening both of Adobe's principal revenue-generating products. Future versions of Apple's LaserWriter series will be equipped with Microsoft's PostScript-clone interpreter firmware. Imagesetter manufacturers will be forced to support both Adobe and Microsoft/Apple PostScript imaging models. Software developers will also be compelled to follow market trends. While these events do not bode well for the prospect of a standardized graphics file format, commentators have predicted that "PostScript can be expected to remain an industry standard." Instead of one unified format, "look for two font and page-description standards coexisting with printer manufacturers generally supporting both" (Anderson 1990).

Predictions of this sort seemed to have been fulfilled in agreements reached between Apple and Adobe at the 1990 Seybold conference. While terms of the agreement have not been officially disclosed, one report suggests that ". . . Apple will make it easy for Adobe fonts to be used with System 7 Macintosh operating software, and Adobe will make it easy for TrueType fonts to be used with its PostScript printer language" (Lewis 1990).

Because the products we create are diverse, microcomputer cartographers must contend with a variety of software packages and digital file formats. I hope the information assembled above proves useful to new users who inevitably will be confronted with this knotty problem. It would be nice to see a comparable note for IBM-compatible PC file formats in a subsequent *CP.* Φ

ACKNOWLEDGMENTS

The comments of Tony Canike (Accu-Weather), John Krygier (Penn State), Bill Peterson (Penn State) and three anonymous reviewers are appreciated.

REFERENCES

Adobe Systems, Inc. (1985) *PostScript Language Reference Manual.* Addison-Wesley.

Adobe Systems, Inc. (1989) Adobe Illustrator™ Document Format Specification (draft). Special Systems Group.

Anders, J.K. and K. Lathan (1990) The ins and outs of CAD data exchange. *Macintosh-Aided Design,* August, pp. 37-45.

Anderson, B. (1990) Unraveling the upheaval under way in PostScript. *Computer Graphics Review,* March, pp. 20-24.

DiBiase, D. Linking illustration and mapping software through PostScript (in preparation).

Electronic Publishing and Printing (1989) 2nd annual service bureau directory. Maclean-Hunter.

Lewis, P.H. (1990) On the battlefront, an uneasy truce raises hopes. *The New York Times,* Sunday, October 14.

OTHER SOURCES

Apple Computer, Inc. (1985) *Inside Macintosh Volume I.* Addison-Wesley.

Apple Computer, Inc. (1986) *Inside Macintosh Volume V.* Addison-Wesley.

Hewlett-Packard Company (1987) A guide to the Tag-Image File Format.

McClelland, D. and C. Danuloff (1989) *Mastering Aldus FreeHand.* Dow Jones-Irwin.

Parascandolo, S. (1989) Putting it graphically. *MacUser,* March, pp. 189-194.

Parascandolo, S. (1990) Drawing the line. *MacUser,* April, pp. 93-107.

Roth, S.F. (1988) *Real World PostScript.* Addison-Wesley.

Standardized PostScript Association (198?) Renewing the promise of PostScript. Unpublished manifesto.