U.S. GEOLOGICAL SURVEY NATIONAL COMPUTER TECHNOLOGY MEETING: PROCEEDINGS, PHOENIX, ARIZONA, NOVEMBER 14-18, 1988

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 90-4162

U.S. GEOLOGICAL SURVEY NATIONAL COMPUTER TECHNOLOGY MEETING: PROCEEDINGS, PHOENIX, ARIZONA, NOVEMBER 14-18, 1988

By Barbara H. Balthrop and John E. Terry, editors

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 90-4162

science for a changing world

Click here to return to USGS Publications

Nashville, Tennessee

1991

U.S. DEPARTMENT OF THE INTERIOR
MANUEL LUJAN, JR., Secretary

U.S. GEOLOGICAL SURVEY
Dallas L. Peck, Director

For additional information write to:

Coordinator, National Computer Technology Meeting Proceedings 1988
U.S. Geological Survey
300 W. Congress
Federal Building, FB-44
Tucson, Arizona 85701

Copies of this report can be purchased from:

U.S. Geological Survey
Books and Open-File Reports Section
Federal Center
Box 25425
Denver, Colorado 80225

FOREWORD

The U.S. Geological Survey National Computer Technology Meetings (NCTM) are sponsored by the Water Resources Division and provide a forum for the presentation of technical papers and the sharing of ideas or experiences related to computer technology. This report serves as a proceedings of the meeting held in November, 1988 at the Crescent Hotel in Phoenix, Arizona. The meeting was attended by more than 200 technical and managerial people representing all Divisions of the U.S. Geological Survey.

Scientists in every Division of the U.S. Geological Survey rely heavily upon state-of-the-art computer technology (both hardware and software). Today the goals of each Division are pursued in an environment where high speed computers, distributed communications, distributed data bases, high technology input/output devices, and very sophisticated simulation tools are used regularly. Therefore, information transfer and the sharing of advances in technology are very important issues that must be addressed regularly.

This report contains complete papers and abstracts of papers that were presented at the 1988 NCTM. The report is divided into topical sections that reflect common areas of interest and application. In each section, papers are presented first followed by abstracts. For these proceedings, the publication of a complete paper or only an abstract was at the discretion of the author, although complete papers were encouraged.

Some papers presented at the 1988 NCTM are not published in these proceedings.

John E. Terry

CONTENTS

CHAPTER E--DATA BASES AND AUTOMATED DATA HANDLING

CHAPTER F--ELECTRONIC PUBLISHING/MANUSCRIPT PREPARATION

CONVERSION FACTORS

Multiply	By	To obtain
inch (in.)	25.4	millimeter
inch per day (in/d)	25.4	millimeter per year
foot (ft)	0.3048	meter
foot per day (ft/d)	0.3048	meter per day
foot per mile (ft/mi)	0.1894	meter per kilometer
cubic foot per second (ft^3/s)	0.02832	cubic meter per second
mile (mi)	1.609	kilmometer
square mile (mi^2)	2.590	square kilometer
mile per square mile (mi/mi^2)	0.621	kilometer per square kilometer
gallons per day (gal/d)	0.003785	cubic meter per day
million gallons (Mgal)	3,785	cubic meter
million gallons per day per square mile [$(Mgal/d)mi^2$]	1,460	cubic meter per day per square kilometer

Temperature in degrees Celsius (°C) may be converted to degrees Fahrenheit (°F) as follows:

$$°F = 1.8 \times °C + 32$$

The use of brand, company, or trade names in this report is for indentification purposes only and does not constitute endorsement by the U.S. Geological Survey.

CHAPTER A--USGS COMPUTER MANAGEMENT/ADMINISTRATION

MAGNETIC-TAPE BACKUP AND ROUTINE MAINTENANCE PROCEDURES FOR A MINICOMPUTER SYSTEM OF THE U.S. GEOLOGICAL SURVEY

By John E. Owen

U.S. Geological Survey

ABSTRACT

The Arkansas District of the U.S. Geological Survey relies on the dependability and efficiency of a minicomputer to process basic data, conduct interpretive studies, and to meet the needs of cooperators. Hydrologic data, processing programs, and user information on the minicomputer disk partitions are maintained in an accurate, logical, and secure manner. The procedures used to perform the daily, biweekly, and monthly backups as well as the procedures used for pulling, cleaning, and testing magnetic tapes are discussed in this paper. The schedules for performing magnetic-tape backup operations and routine maintenance procedures for the minicomputer also are discussed. A minimum of downtime and virtually no loss of data demonstrates the effectiveness of this operation in the Arkansas District.

INTRODUCTION

The Arkansas District of the U.S. Geological Survey uses the PRIME minicomputer and its capability to respond to the needs of the district and its cooperators. The district maintains information in an accurate, logical, and secure manner with a minimum amount of downtime. This paper describes the steps, programs, and files used to perform daily (incremental), biweekly, and monthly magnetic-tape backups; the procedures used for pulling, cleaning, and testing magnetic tapes; and the schedule for performing routine maintenance of the minicomputer.

Cleanliness and controlled climate are essential to dependability of hardware and efficiency of the minicomputer. All tapes and spare disk packs are maintained in the controlled environment of the computer room or in an offsite, climatically-controlled vault. Limited access is allowed in these areas. All cleaning of the controlled area is accomplished by the computer operator.

MAGNETIC-TAPE BACKUP PROCEDURES

Daily, biweekly, and monthly magnetic-tape backups are performed with the PRIMOS utility program (MAGSAV), which is monitored by a magnetic tape backup log and controlled by a locally written menu driven Command Procedure Language (CPL) program and executed by a Command Input (COMI) file. The Command Procedure Language makes use of such "high-level language" features as branching and argument transfer to simplify and automate long command sequences and to allow decision making at the command level. The COMI file controls the flow of commands to the CPL, insuring that logical tape numbers will be standardized, providing expedient access in the event that file recovery becomes necessary. Index files also are maintained and stored for all backup tapes to quickly locate the tape that contains any on-line file that is lost or destroyed. The appropriate tape is then used to restore the damaged or destroyed file. This practice has virtually eliminated loss of data.

Daily Backup

Daily backup tapes are maintained for 90 days. When the 90-day period has been exceeded, the tapes are pulled from the tape rack and the labels removed. The tapes are then processed through a tape cleaner before being returned to the computer room for use. All tapes are kept on a tape rack in a climatically-controlled room. Only tapes that have been in this environment for at least 24 hours are used. Tapes are not stored lying flat even when waiting to be mailed or shipped to offsite storage.

Biweekly and Monthly Backup

Biweekly and monthly backups are performed on Sunday afternoons. This time has been selected because it does not interfere with normal operations and reduces the amount of down time. These backups, like the daily backups, are performed by the MAGSAV utility program, which is controlled by a CPL program. Biweekly backups are done on the first day of each pay period. The purpose of this date is to capture all data for the prior period. It is also easier for an employee to determine what day the computer will be unavailable. Also, long running models or other batch jobs have time to complete before the biweekly backups are done on Sunday. The biweekly backup is executed with all system processes shut down and only the system is running. With priority access rights set for the system, every file is backed up. The same backup CPL program that is used for incrementals is used for the total backup and an index for each partition is created. Because of the large volume of records on disk, all tapes are created at a density of 6,250 bits per inch (BPI) which permits a faster completion time and minimizes the number of tapes that are used for backup. Because cold starts from tape require the command device tape to be written at 1,600 BPI, a spare disk pack is stored that can be loaded while maintenance is being performed on the damaged device. Biweekly tapes are stored on a tape rack, and are maintained for 6 months before being pulled, cleaned, and randomly tested before put back in use. These tapes are maintained in date and Master File Directory (MFD) order so that a complete set of backup tapes can be easily located by the operator if a file or directory restoration is necessary.

Monthly backups are done by the same procedure as the biweekly backups, but two copies of each partition are created. One copy stays on a rack designated for monthly backup tapes. The other goes to an offsite storage area that is under contract and maintained under strict access. The offsite tapes are kept in a climatically-controlled vault and can be delivered back to the computer room on request. Monthly tapes are held for 1 year but, rotation of tapes within the cartridge seal is done every 90 days. Higher quality tapes are selected for monthly backups because of the importance of these master files. Cleaning and testing after 1 year is accomplished before tapes are released for reuse.

ROUTINE MAINTENANCE PROCEDURES

Preventive maintenance on the minicomputer is performed by the field engineer on a monthly basis. At this time filters are cleaned or replaced and logs checked for disk errors. Because maintenance visits require the system to be shutdown, they are scheduled during lunch to minimize the amount of time that the machine is unavailable to users. Other routine maintenance operations are performed by the system operator at hours convenient to the user community because most routine maintenance only takes a few minutes. After the biweekly backups are completed, another COMI file is run which executes the PRIMOS disk maintenance utility program, FIX_DISK, which helps insure the integrity and accessibility of all files on the disk.

The minicomputer has the ability to support a remote systems console. When power fluctuations occur at night or on weekends, the computer can be cold started, FIX_DISK can be run to repair damaged files, and phantoms that run continuously can be started from the remote systems console. This convenience can save much travel time and also makes it possible to bring the system up when it might otherwise be down for several hours. A COMI file has been written to perform the FIX_DISK utility and a log is kept so that disk maintenance can be performed to minimize human error. Routine disk maintenance is completed after each biweekly backup in addition to unscheduled maintenance that is performed after power outages and so forth.

After disk maintenance is performed, the log is checked for any disk problem before restarting system phantoms. If a problem exists, the operator will determine if FIX_DISK is to be re-run or if the system administrator is to be notified of the problem. Once maintenance is determined to be complete and all disk files are usable, a COMI file is started to bring up system phantoms.

SUMMARY

The Arkansas District of the U.S. Geological Survey relies on the dependability and efficiency of a minicomputer to process data, do interpretive studies, and meet the demands of its cooperators. With the use of the minicomputer software packages, locally written CPL programs, and COMI files, maintenance and operations of the minicomputer system are accomplished efficiently. A minimum of downtime and virtually no loss of data demonstrates the effectiveness of this operation in the Arkansas District.

SELECTED REFERENCES

Alley, S.A., 1984, Magnetic tape user's guide: Natick, Massachusetts, PRIME Computer, Inc., chapter 9.

-----, 1984, Operator's guide to system backups: Natick, Massachusetts, PRIME Computer, Inc., chapters 1, 5, 6, 8, and 14.

Forbes, J., Landy, A., and Miles, C., 1986, Operator's guide to system commands: Natick, Massachusetts, PRIME Computer Inc., p. 2-40, 2-73, 2-9, and 2-10.

Froenlich, A.F., 1982, Managing the data center: Belmont, California, Wadsworth Inc., Lifetime Learning Publications, p. 19-21, 254.

Gaither, N., 1984, Production and operations management: New York, CBS College Publishing, Dryden Press, p. 710-724.

Gove, G.W., 1984, Operator's guide to file system maintenance: Natick, Massachusetts, PRIME Computer, Inc., chapter 3.

THE DEVELOPMENT OF DISTRIBUTOR SOFTWARE FOR TRANSMITTING DOCUMENTS THROUGH A COMPUTER NETWORK

by
Steven J. Brady[1]

ABSTRACT

The use of electronic mail by the Water Resources Division of the U.S. Geological Survey has evolved from informal use into a system designed to replace paper correspondence within about 5 years. A critical component in developing such a system is distributor software that accomplishes three tasks: (1) automatically verifies the recipients' addresses, (2) uses minimal computer resources, and (3) generates minimal network traffic in a distributed environment.

A distributor is an active process able to recognize certain user activities and to respond on them. In the case of electronic mail, the distributor processes messages being sent as well as those being received. Thus, the distributor allows any user to send electronic messages to any other user.

A distributor process called MAILMAN, written in FORTRAN 77 programming language, has been developed for the Division. This process, which executes in the background, is installed on all Division Prime computers, and is part of a larger system termed Electronic Documents (EDOC). When executing, the program distributes all electronic mail messages and notifies individual recipients of those messages.

Three critical steps were taken in development of the distributor. First, a data base of all electronic mail users in the Division, which automatically verifies a recipient's user identification and location, was created. In addition, group identifiers were assigned to groups of users at each site who share a common speciality and facilitate the sending of messages to thousands of recipients. Second, an algorithm was developed to transfer only one copy of a message to a site regardless of the number of recipients at that site; this maintains low network overhead. Third, the distributor was designed to use a semaphore utility, which assures that the distributor process would work only when notified by a user; this minimizes computer resource usage by the distributor process, while allowing for extremely fast response to the user.

[1] Hydrologist, U.S. Geological Survey, 1400 Independence Road., Rolla, MO 65401

CHAPTER B--DIS/DISTRIBUTED ENVIRONMENT

SUPPORTING DIFFERENT TYPES OF TERMINALS IN A DISTRIBUTED-INFORMATION ENVIRONMENT

By J. W. Atwood and S. D. Bartholoma

ABSTRACT

The Distributed Information System (DIS) of the U.S. Geological Survey has substantially increased computing capabilities in District offices. Program implementation and data entry are no longer done by keypunching cards and developing ordered decks of interspersed instruction and data cards. Instead, programs are created and run, and data is entered interactively using video display terminals (VDT).

Programs have been written to allow data entry by "filling in the blanks" on a formatted VDT screen. Other programs use certain areas of the VDT screen to display messages. This has made computer data entry more "user friendly"; it also has caused some major problems.

Because each District office has been responsible for obtaining its own terminals, different types of terminals are used, sometimes in the same District office. In addition, some District offices also provide cooperating agencies with access to the computer; these cooperators commonly have terminals that differ from those used by the Survey. Problems result when a program written specifically for one type of terminal is run on a different type of terminal. In many cases, the program does not run on the terminal it was not written for. To compound the problem, standard cursor control subroutines do not currently (1988) exist.

With the distribution of common computer programs to all U.S. Geological Survey District offices, it is important to provide programs that are compatible with many types of terminals and to provide the ability to easily add new types of terminals as the District offices acquire them.

A cursor control subroutine library package called the DIS CURsor control package (DISCUR) has been developed, which allows any number of terminals to be supported. A set of standard subroutines has been developed, and new types of terminals can access all programs using DISCUR at the same time, and without re-compilation of the program. The use of DISCUR allows the simplest cursor control or complete control of data output and input. This report documents the development of the DISCUR control package; the report is not intended to be a user's guide.

INTRODUCTION

In late 1982 and early 1983, the U.S. Geological Survey installed a nationwide network of about 70 multi-user minicomputers. This network, known as the Distributed Information System (DIS), has substantially increased the computing capabilities available at the District-office level. Program implementation and data entry are no longer done by keypunching cards and developing ordered decks of interspersed instruction and data cards. Instead, programs are created and run, and data is entered interactively using video display terminals (VDT).

The purpose of this report is to document the development of a cursor control package designed to assist computer programmers in writing programs that contain "user-friendly" input and output functions. This report is not intended to be a user's guide.

DESCRIPTION OF PROBLEM

Programs have been written to allow data entry by "filling in the blanks" on a formatted VDT screen. Other programs use certain areas of the VDT screen to display messages. This has made computer data entry more "user friendly"; it also has caused some major problems.

Because each District office has been responsible for obtaining its own terminals, different types of terminals are used, sometimes in the same District office. In addition, some District offices also provide cooperating agencies with access to the computer; these cooperators commonly have terminals that differ from those used by the Survey. Problems result when a program written specifically for one type of terminal is run on a different type of terminal. In many cases, the program does not run on the terminal it was not written for; the terminal may lock up and no longer accept keyboard commands. An unreadable screen may be displayed or a somewhat readable screen may be displayed, but the cursor may not be positioned correctly to indicate the required user response. To compound the problem, standard cursor control subroutines do not currently (1988) exist.

With the distribution of common computer programs to all Geological Survey District offices, it is important to provide programs that are compatible with many types of terminals and to provide the ability to easily add new types of terminals as District offices acquire them.

DESIRED SOLUTION

To solve the problems of using different types of terminals, a screen-control subroutine library with the following characteristics is needed:

1. The library needs to support as many types of terminals as possible.

2. Screen-control features need to be accessed by a set of standard subroutines. Programs need to be independent of the type of terminal used, with the exception of optionally requiring cursor-control and screen-control capabilities.

3. Programs using the screen-control library should not have to be re-compiled or re-loaded when a terminal is modified or a different type of terminal is added to the system.

4. The screen-control library needs to be installed as a stand-alone library at the operating-system level. It needs to be available to all programmers; it also needs to be compatible with all other programs in the computer, but not be an integral part of these programs.

5. The library needs to record the type of terminal used by a particular user, so that the user does not have to specify the terminal type every time the library is invoked.

EXISTING PROGRAMS

Several programs for terminal control sequences have been developed and used by U.S. Geological Survey and others. Some of them are described below.

Hard-Coded Control Sequences

Many applications in the DIS have been implemented using hard-coded, terminal control sequences. These applications, written by various programmers within the Geological Survey, support terminals using the American National Standards Institute (ANSI) terminal control sequences. The Administrative and FInancial data Management System (AFIMS) program uses this technique. Programs with hard-coded ANSI sequences can only be used on ANSI terminals such as the Digital Equipment VT100 and its equivalents (TAB, Lear-Seigler ADM36, Graphon, Tektronix, and others).

SCREEN Program

The SCREEN program was written in the New Jersey District office of the Geological Survey in early 1980 by Stephen M. Crutchfield, a student from Drexel University. This program supported several different terminal types through the inclusion of a separate subroutine for each type of terminal. Identification of the specific type of terminal to the program was through a variable in a Fortran common block.

Adding a new type of terminal necessitated writing a new driver subroutine for that type of terminal and adding a command to invoke the new subroutine in the SCREEN subroutine. The new driver subroutine and the SCREEN subroutine then had to be re-compiled, and all programs that invoked SCREEN had to be re-loaded to make the new type of terminal available to users.

SCREEN was used in the Automatic Data Recorder (ADR) program that was installed in 1983 by the Geological Survey as an interim records-processing package. Re-named S_SCRN, it was later used in the original version of the Automatic DAta Processing System (ADAPS), which was released in 1986.

CURCON Program

The CURCON subroutine library was written by John W. Atwood while working for the University of Utah Research Institute/Earth Science Laboratory (UURI). Programming for the first revision of CURCON was finished in January 1981. Like SCREEN, this library supported several types of terminals by including a separate subroutine for each type of terminal. The type of terminal was specified by calling a terminal-selection subroutine that identified the selected type of terminal to the program through a variable in a Fortran common block.

Like SCREEN, adding a new type of terminal meant writing a subroutine containing the necessary control sequences and placing commands to invoke the new subroutine in the initialization and driver subroutines. The CURCON subroutines then had to be re-compiled, and all of the programs using CURCON had to be re-loaded.

CURCON was used in several programs written at UURI. Later, CURCON was used in a screen entry program for creating data entry files for the Ground-Water Site Inventory (GWSI) data base on the Survey's mainframe computer in Reston, Virginia. CURCON was also used in the SOFTware EXchange (SOFTEX) program and in the GWSI and Quality of Water (QW) parts of the National Water Information System (NWIS).

TERMINALID Program

As part of the ADAPS system, the TERMINALID program was developed by Joseph Riggsbee in the North Carolina District of the Geological Survey, with the help of Scott D. Bartholoma, Utah District office. This program maintained a data base of type of terminals keyed by user number. ADAPS applications could get a user's type of terminal without having to query the user every time the program was used. The program was later extended to include type of terminals for other, non-Survey software packages such as INFO, EMACS, and TELL-A-GRAF. This program was embedded in the NWIS software structure and was not readily available for non-NWIS applications.

IMPLEMENTED SOLUTION

To remedy the shortcomings of the existing cursor-control programs, the DISCUR subroutine library package was written by Scott D. Bartholoma (Utah District office of the Geological Survey) and John W. Atwood (North Dakota District office). The DISCUR library provides a set of standard cursor-control functions that can be used in screen-entry and tabling programs. Support for new terminals can be added by using a definition builder to create a new terminal definition and by adding a one-line description of the terminal to a sequential file. Any program that uses the DISCUR library for cursor control will work properly on the newly added terminal without re-compilation of the program.

The DISCUR library consists of three subpackages. The first subpackage, called TERMINALID, keeps track of the user's type of terminal. It consists of a set of operating-system-level commands and a set of user-invokable subroutines to display, select, set, and retrieve terminal-type information.

The second subpackage, referred to as the C_SUBS subroutines, is used for simple cursor control only. The names of these subroutines begin with "C_", and each subroutine performs a single function. This set of subroutines may be the best choice if a user wishes to do only simple cursor control because there is no additional storage of cursor position or screen image.

The third subpackage, called CURCON, provides terminal cursor control as well as screen-entry and field-editing capabilities. The cursor position as well as a memory image of the current screen are maintained by this group of subroutines. This set of subroutines can be used when complete control of the terminal screen is desired.

Cursor control in DISCUR is accomplished through the use of definition files. When a program using DISCUR is used, a subroutine is invoked that loads the control-code sequences into the package from the definition file of the specified terminal.

Installation of a new terminal in the DISCUR cursor-control package is relatively easy, requiring only the creation of a definition file and the addition of a line to a sequential file containing a list of available types of terminals. No re-compilation is required.

Defining New Types of Terminals

A definition file is created with the definition-building program (BLDDEF), which creates an American Standard Code for Information Interchange (ASCII) definition file, and compiles the ASCII file into an unformatted, binary, direct-access file containing the control-code sequences to perform individual cursor movements and functions. The user's manual for the terminal usually contains a list of the control-code sequences needed for entry into the BLDDEF program.

All ASCII definition files are given the name of the terminal suffixed with ".SRC". An ASCII definition file is created by the BLDDEF program whenever the specified file name does not exist. After the ASCII definition file has been created using BLDDEF, the file can be modified using any text editor and then re-compiled by running BLDDEF again. The resulting compiled file is given the name of the terminal suffixed with ".CFG".

The subroutines C_INIT and SELTER are used to load the binary definition file into the control-code storage area. The control-code storage area is a section of data storage, internal to the DISCUR package, that contains all the control-code sequences loaded from the binary definition file. The control-code storage area provides the corresponding driver subroutines with the control-code sequences for the specified terminal. C_INIT is for use with the C_SUB group of subroutines, and SELTER is for use with the CURCON group of subroutines.

TERMINALID Subpackage

Programs that use the DISCUR library, as well as many other programs on the Survey's minicomputers, use cursor control and require the type of terminal to be specified. It is convenient for users not to have to specify the type of terminal every time a program using cursor control is invoked. The TERMINALID subpackage provides a mechanism for the user to specify the type of terminal being used, and the information is stored for later use.

The TERMINALID package consists of two data files, three system-level commands, some utility programs, and three user-invokable subroutines. A sequential file, named DISTRICT_TERMINALS, contains an entry for each type of terminal available and, in each entry, some descriptive information and the terminal identifiers to be specified for DISCUR and for several other proprietary programs. This is an ASCII file and may be edited to add new types of terminals and to indicate types of terminals that are available at the local computer site.

A direct-access file, named USER_TERMINALS, contains each user's terminal information filed by user number. A user on a hard-wired terminal line (as most users in the Geological Survey are) can specify the type of terminal once and change it only if another type of terminal is attached to the line. A command suitable for inclusion in a login-time command file to set the type of terminal is provided for users of a network or port-sharing device where a pool of user numbers is shared.

System-Level Commands

Three system-level commands are associated with the TERMINALID package:

1. The TERMNL command is used to identify the type of the user's terminal for applications using the DISCUR cursor-control package and for other applications where the user's type of terminal is needed.

2. The SHOWTERM command displays the currently selected type of terminal.

3. The TERMTYPE command, which also can be invoked as a command function, extracts the type of terminal for DISCUR or any one of several other proprietary software packages, and returns it for later use.

Utility Programs

When it becomes necessary to modify some of the information in the DISTRICT_TERMINALS file, the modifications are not automatically forwarded to users who have selected the modified type of terminal. A utility program to forward the new information to the USER_TERMINALS file, named RELOAD_TERM, has been provided.

Subroutines

Three subroutines are provided as part of the TERMINALID subpackage that duplicate the functions of the three commands described earlier. The subroutines are:

C_SETT - Sets the type of terminal for a user;
C_TMSG - Displays the user's current type of terminal; and
C_TTYP - Retrieves terminal information.

C_SUBS Subpackage

The C_SUBS subpackage consists of a set of subroutines used to move the cursor, erase all or part of the terminal screen, set character and line attributes, and other miscellaneous terminal-control functions. These subroutines retrieve the control sequences for the various functions from the control-code storage area that has been loaded by the initialization subroutine. The contents of the control-code storage area are loaded from a binary file that is created by the DISCUR utility "BLDDEF". If a function is not supported by the user's terminal, the subroutines return without taking any action. Following is a list of the subroutines available in the C_SUBS subpackage, grouped by function.

Initialization is controlled by the subroutine C_INIT. It retrieves the DISCUR driver name from the TERMINALID package and loads cursor-control information. The programmer may specify that only terminals that support cursor control may be used. If the selected terminal does not support cursor control, an appropriate code is returned. This subroutine needs to be used before any other C_SUBS cursor-control subroutines are used.

These subroutines control cursor movement:

C_UP - Moves cursor up;
C_DOWN - Moves cursor down;
C_LEFT - Moves cursor left;
C_RGHT - Moves cursor right;
C_POSN - Moves cursor to a specified location; and
C_HOME - Moves cursor to "home" position (top left corner).

These subroutines control screen erasure:

C_EBOL - Clears from beginning of line;
C_EEOL - Clears to end of line;
C_ELIN - Clears entire line;
C_EBOS - Clears from beginning of screen;
C_EEOS - Clears to end of screen;
C_EALL - Clears entire screen, cursor is not moved; and
C_CLS - Clears entire screen and moves cursor to "home" position.

These subroutines control screen attributes:

C_DB - Bold;
C_DBF - Bold, flashing;
C_DBFR - Bold, flashing, reversed;
C_DBR - Bold, reversed;
C_DBU - Bold, underlined;
C_DBUF - Bold, underlined, flashing;
C_DBUR - Bold, underlined, reversed;
C_DU - Underlined;
C_DUF - Underlined, flashing;
C_DUFR - Underlined, flashing, reversed;
C_DUR - Underlined, reversed;
C_DF - Flashing;

C_DFR - Flashing, reversed;
C_DR - Reversed;
C_DALL - Bold, underlined, flashing, reversed; and
C_DOFF - All attributes off, normal characters.

These subroutines control line attributes:
C_LNDT - Double width, double height, top one-half;
C_LNDB - Double width, double height, bottom one-half;
C_LNDW - Double width, single height; and
C_LNOF - Attributes off - Single width, single height.

These subroutines are used for miscellaneous functions:
C_LOCK - Lock lines;
C_SCRL - Set scrolling region;
C_NARO - Set to narrow screen;
C_WIDE - Set to wide screen;
C_AXON - Auxiliary (printer) port on;
C_AXOF - Auxiliary (printer) port off;
C_BELL - Ring terminal bell; and
C_RSET - Perform master reset.

CURCON Subpackage

CURCON is the primary subroutine within the CURCON subpackage. CURCON is used to perform all of the cursor-positioning functions. Other subroutines are used to: load the control-code storage area with the proper cursor-control sequence codes (SELTER); select an option (LISOPT); enter data (REDSTR, REDTMP, REDNUM, REDDAT, REDTIM, REDSCR); display an error message (MESQUR); and access individual values within the control-code storage area (RTRVPR, STROPR).

For terminals that support only cursor positioning, CURCON can emulate most of the other clearing and scrolling functions. Emulation of the clearing and scrolling functions requires strict control of the cursor position. This is why the cursor position and screen image are stored when using CURCON. The mixing of other screen control subroutines with CURCON, although possible, is not recommended because they modify the screen image and move the cursor without updating the stored screen image and cursor position variables. Fortran "read" and "write" statements used to perform terminal input and output should also be avoided when using CURCON, for the same reason. All output to the screen needs to be through the CURCON, MESQUR, and LISOPT subroutines. All entry from the terminal needs to be through one or more of the REDxxx subroutines. Otherwise, the program will not work properly on terminals that require the emulation of certain functions.

The subroutine SELTER needs to be used to initialize the control-code storage area before any of the other CURCON subroutines are used. SELTER has no arguments. The type of the user's terminal is read from a system file that first needs to be set using the TERMNL command at the operating-system level. If the type of the user's terminal is undefined, SELTER will display a list of the available type of terminals and prompt the user to select one. The list of terminals given is dependent both on the DISTRICT_TERMINALS file and on whether the user is a local or remote user. If the user is on a hard-wired line, only the terminals that have been defined for the local office are listed. If the user is a remote user, the entire list of available terminals is displayed.

CURCON is used to perform all cursor positioning and attribute setting. Areas within the screen can be blanked, filled, or edited. A scrolling region can be set in both the vertical and horizontal directions. Messages are displayed in normal, bold, underlined, reverse video, and double-sized characters if the terminal supports such attributes.

LISOPT is used to display an option list and accept a decision on the desired option.

MESQUR is used to display an optional message at the bottom of the screen and pause for permission to continue. If the user's program detects an error, a message can be displayed using MESQUR. The message is displayed on the last line, and a prompt to continue is displayed on the line above the message. Special messages and prompts can be displayed in the same manner. MESQUR also displays a two-line command description at the bottom of the screen. The description lines are displayed on the same two lines used to display messages, and replace the messages after exiting the MESQUR subroutine. If desired, the command description lines can be displayed initially or re-stored after a screen clear by using a special command to MESQUR. The command description lines contain a visual reminder of a few of the field-editing commands available to the user. These field-editing commands are used by the data-entry subroutines, as well as LISOPT and MESQUR, to perform within-field editing.

The data-entry subroutines REDSTR, REDTMP, REDNUM, REDDAT, REDTIM, and REDSCR are used for entering data of various types. All data needs to be entered using these subroutines so that the program can always maintain a record of the cursor location. With the exception of REDSCR, the data-entry subroutines allow within-field editing. REDSCR is a special subroutine that reads a field of specified length directly from the stored screen image. All entered data is in character-string form. If the entered data is a numerical value, it should be read first by REDNUM, and then an internal read (from a character variable, instead of from a file) should be performed on the character string to retrieve the required numeric value. The field-editing codes that define field positioning and edit modes are passed to the invoked subroutine for proper execution. This allows the invoked subroutine to position to the proper field, change to edit or entry mode, store the data, or abort.

The last subroutines, RTRVPR and STORPR, are used to access the individual values in the DISCUR control-code storage area. Care needs to be taken in using these subroutines, as erroneous modification of the DISCUR system variables can have unexpected results. These subroutines are provided so that programmers can access the values contained in the control-code storage area. These subroutines give direct access to the control sequences as well as the stored screen.

CONCLUSIONS

With the increasing use of VDT terminals for entry of data to a computer, there is a need for the use of standard terminal-control subroutines. The need for supporting many types of terminals is great. The DISCUR control package has been developed to fit these needs. DISCUR provides a complete set of terminal control features and allows easy addition of any number of types of terminals. Addition of new terminals to the system is independent of any program using the DISCUR package. To use a newly added type of terminal, the user need only select the new terminal before running programs containing commands to DISCUR. DISCUR is a stand-alone package that can be used in any program that performs cursor control.

CHAPTER C--GIS APPLICATIONS IN USGS

Displaying Data from the National Water Data Exchange by Use of a Geographical Information System

Bruce Parks
U.S. Geological Survey
Office of Water Data Coordination
Reston, Virginia

ABSTRACT

The capability exists to plot the location of the water-data-collection sites for each State by using a geographic information system program with appropriate background files. Information from the Master Water Data Index of the National Water Data Exchange, a data base maintained at the U.S. Geological Survey's headquarters office in Reston, Virginia, has been transferred into a series of digital files in a geographical information system that represents features on maps. "Key files" and special commands allow rapid selection of records by type of data collected, collecting agency, frequency of data collection, or any combination of selection criteria. The relation and structure of the files and practical applications in which they could be used are examined.

BACKGROUND

The Office of Management and Budget Circular A-67 gives the Department of the Interior the responsibility to maintain a catalog of water-data information on a national basis. The National Water Data Exchange (NAWDEX) is an interagency program residing in the U.S. Geological Survey to facilitate the exchange of water data and to promote the standardization of procedures for handling water data. Federal, State, local government organizations, academic institutions, and private organizations that collect and use water data are participants in the NAWDEX program. NAWDEX maintains a Master Water Data Index (MWDI), which is a computerized index of available water data. This index resides on an Amdahl mainframe computer at the Geological Survey Headquarters office at Reston, Virginia.

Requests for information from NAWDEX may be made through the 76 NAWDEX Assistance Centers located throughout the country. Responses to requests are usually given in tabular form, although limited capability exists for showing the areal distribution of sites. For the novice user of NAWDEX, retrievals can be time consuming and may require resubmittal to get the exact information or format required. The purpose of this paper is to describe a scheme for transferring data from NAWDEX to a geographical information system that can be used interactively to display the information on maps, and to identify some applications for using the system.

Water-resources managers and data users need to be able to readily see the data that are available for their area of interest. This is best accomplished by using a computerized information management system that allows the display of information on maps and modification of the information being displayed. ARC/INFO, a Geographical Information System (GIS) developed by the Environmental Systems Research Institute of Redlands, California, is a management tool that can be used to organize, manipulate, and graphically display the geographic data available in the MWDI.

APPROACH

To display the location of sites listed in the MWDI, data files need to be developed that contain the information in the MWDI and reside in the data base structure of a GIS. Software has been developed that will read all information in the MWDI for a set of sites and write the information to tape. The tape is then loaded onto a Prime minicomputer where a program edits and formats the file, builds a data base, and creates an ARC/INFO coverage, which is a digital analog of a single map sheet stored in a suite of files. INFO is a relational data base software program developed by HENCO Software, Inc., Waltham, Massachusetts, which is used to build the data base. ARCPLOT, a subsystem of ARC/INFO, is then used to display the location of the sites or subsets of sites along with other geographical data showing State and county boundaries, hydrologic units, river locations, and other information, that has been drawn from other data files. The subsets can be controlled by selecting a range of values for any of the parameters or items listed for the sites.

Structure of the Master Water Data Index

The MWDI data base on a mainframe computer is maintained through a data base management system called SYSTEM 2000. The data in the SYSTEM 2000 data base are organized into a hierarchical structure as shown in figure 1. The MWDI contains the following general categories of information:

Station Identifiers

- Unique identifiers (site number, agency number, NAWDEX number)
- Operating organization
- Geographical identifiers (country, State, county, hydrologic unit)
- Type of site (stream, well, lake, spring, estuary)
- Physical identifiers (drainage area, basin descriptor)
- Station status (active, inactive)
- Supplementary data available

Type of water data collected

- Surface water
- Ground water
- Water quality
- Meteorological

Attributes of water-data collected

- Period of record
- Record continuity
- Data parameters collected
- Frequency of data collection
- Data storage media
- Purpose of activity
- Status of activity

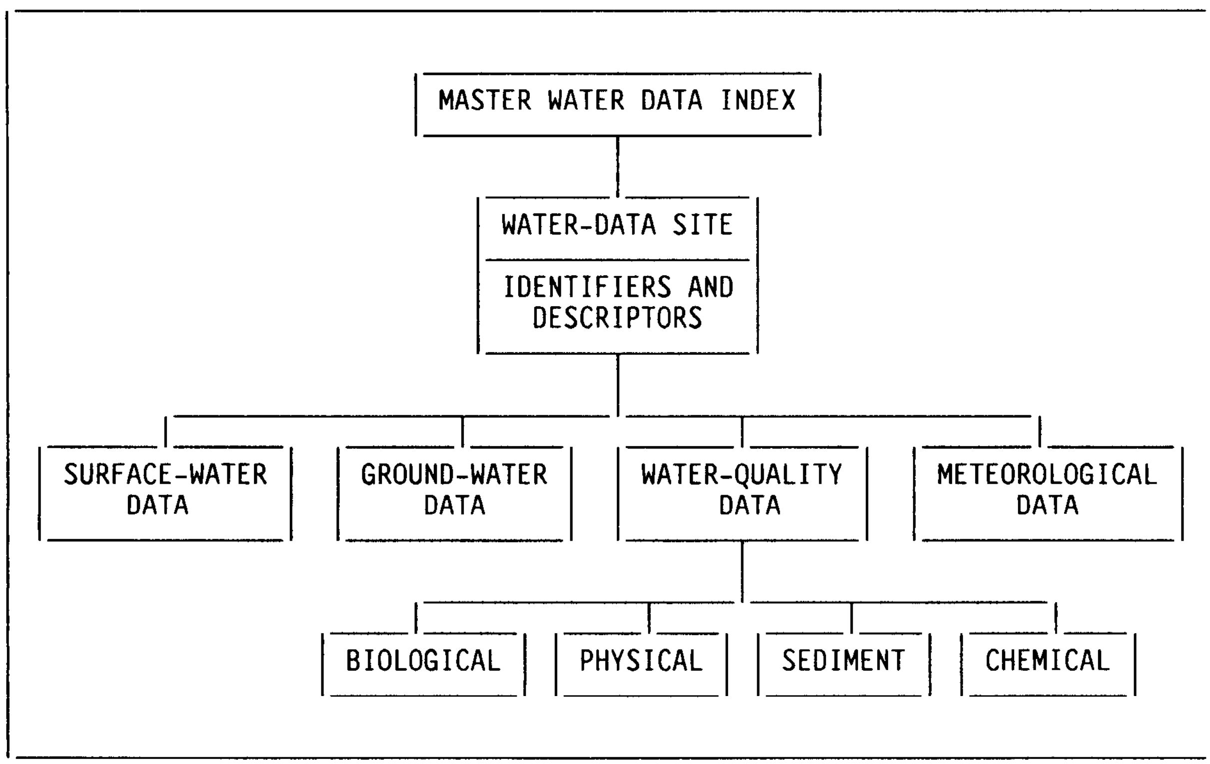


Figure C1. Structure of the Master Water Data Index's SYSTEM 2000 data base.

Transfer of Data from the National Water Data Exchange to the Prime

The MWDI resides on the Amdahl computer at the Geological Survey in Reston, Virginia. Job Control Language (JCL), a computer software program, was used to make retrievals by State or territory for all available data from each site and to write the information to tape. Prime's magnetic tape utility program, MAGNET, was used to read the file into the Prime environment and the GET COPY command in INFO was used to transfer the data into an INFO file. An ARC/INFO coverage was created by using the GENERATE command in ARC/INFO with the latitudes and longitudes from the MWDI data, and by projecting the coverage into Albers equal-area coordinates.

The MWDI contains more than 200 parameters for each of the 460,000 sites listed for the United States. The data were categorized by State to keep the files in a more convenient size. To maintain continuity between the original data base and the data base to be created for the GIS environment, a parallel system was developed in ARC/INFO. A directory, or suite of files, was created for each State. The component names used in the SYSTEM 2000 data base have been used as item names in the ARC/INFO data base and the groups of components used in the SYSTEM 2000 data base have been organized as separate files in the ARC/INFO data base. The suite of ARC/INFO files that correspond to the structure shown in figure C1 are given in table C1. Directories that contain two sub-directories -- named INFO and LOC -- were established for each State. The INFO directory contains the normal ARC/INFO files plus the separate data files for the types of data available. The LOC directory is the ARC/INFO coverage containing the files that define the attributes or boundaries of points, lines, and polygons in the coverages. Both directories are accessed through the ARC/INFO software.

Application

To display the data, ARCPLOT is used with the new command KEYSELECT, described by Lanfear and Hitt (1988). Key files and lookup tables can be created by using common items such as SW_KEY, GW_KEY, and so forth, found in the LOC.PAT file and KEY-LOC which is found in the other files (table 1). Using the searching techniques described by Lanfear and Hitt (1988), the user can select a group of records based on availability, non-availability, or magnitude of any of the parameters in any of the INFO files in a relatively short period of time.

There are a number of coverages in Geological Survey files that can be used as base maps to display the states, counties, rivers, water bodies, and hydrologic units. The sites selected from the NAWDEX files can be overlayed on these maps to show the spatial distribution of the sites. Because there are codes in the NAWDEX files that list the State, county, hydrologic unit, and names for the sites, discrepancies in the data base can be detected by overlaying sites listed for a certain area on a polygon outlining that area, whether by State, county, hydrologic unit or combination.

SUMMARY

An index of water-data information for the United States currently resides in the U.S. Geological Survey's National Water Data Exchange and is maintained on an Amdahl computer at the Geological Survey's headquarters in Reston, Virginia. To have the information readily available for use in a GIS environment, the data were transferred to a Prime computer, copied into INFO files, and developed into ARC/INFO coverages for use with the ARCPLOT subroutines. The files were set up so that keyfiles and lookup tables could be used to decrease the time required for selecting and reselecting data. Because the individual files can be fairly large, the KEYSELECT command is used to decrease the time for selecting the desired group of records.

TABLE C1. List of items in each file in INFO directory

FILE NAME: LOC.PAT	FILE NAME: SW_DATA	FILE NAME: GW_DATA
AREA	NAWDEX#	NAWDEX#
PERIMETER	SW_BEGIN_YR	GW_BEGIN_YR
LOC#	SW_END_YR	GW_END_YR
LOC-ID	SW_INTERRUPTED	GW_INTERRUPTED
NAWDEX#	SW_OWDC_NO	GW_OWDC_NO
NAWDEX_ID	SW_OWDC_SEQ	PRIN_AQUIFER
NAWDEX_AGCY	COMPLETE	AQUIFER_TYPE
AGCY_STA_NO	PEAK_STAGE	LEVEL_FREQ
STATION_NAME	LOW_STAGE	LEVEL_MED
NON_US_COUNTRY	STAGE_MED	DISCHRG_FREQ
STATE	COMPLETE_FLOW	DISCHRG_MED
COUNTY	PEAK_FLOW	SUBSIDE_FREQ
HYDROL_UNIT	LOW_FLOW	SUBSIDE_MED
CONG_DIST	MISC_FLOW_MEAS	DEPTH_OF_WELL
SITE_TYPE	FLOW_MED	GW_RECMD_MTHDS
BASIN_DESCRP	VOLUME	GW_OTHER
WDSD_OFC_CODE	VOLUME_CHANGE	MAJOR_VAR
DRAINAGE.AREA	VOLUME_MED	GW_TELEMETRY
NC_AREA	SW_UNIT_FLOW	GW_LST_UPDATE
LAST_UPDATE	SW_UNIT_STAGE	GW_PURPOSE
STATE_COUNTY	SW_UNIT_VOLUME	GW_RECORDER_TYPE
PRIMARY_USE	SW_RECMD_MTHDS	GW_RECORDER_FREQ
WRD_ACCT	SW_OTHER	GW_PN_CODE
DOWNSTREAM_ORDER	SW_TELEMETRY	GW_MODIFIERS
OTHER_DATA	SW_LST_UPDATE	KEY-LOC **
SW_ACTIVE	SW_PURPOSE	
SW_KEY *	SW_RECORDER_TYPE	
GW_ACTIVE	SW_RECORDER_FREQ	
GW_KEY	SW_PN_CODE	
QW_ACTIVE	SW_MODIFIERS	
QW_KEY	KEY-LOC **	
BIO_ACTIVE		
BIO_KEY		
PHY_ACTIVE		
PHY_KEY		
SED_ACTIVE		
SED_KEY		
CHM_ACTIVE		
CHM_KEY		
MET_ACTIVE		
MET_KEY		
MISC_INFO_KEY		

* Physical record number of corresponding record in the SW_DATA file, for use with INFO's RELATE...LINK command. Other KEYs are similar. Not all stations in the LOC.PAT file have corresponding records in the QW_DATA, GW_DATA, or similar files; KEYs for these records are set to ZERO.

** Physical record number of corresponding record in the LOC.PAT file.

TABLE C1. List of items in each file in INFO directory -- Continued

FILE NAME: QW_DATA	FILE NAME: CHM_DATA	FILE NAME: BIO_DATA
NAWDEX#	NAWDEX#	NAWDEX#
QW_BEGIN_YR	SOLIDS_DIS	ENTERIC_BACT
QW_END_YR	MAJOR_IONS	NATIVE_BACT
QW_INTERRUPTED	HARDNESS	PHYTOPLANKTON
QW_OWDC_NO	SILICA	ZOOPLANKTON
QW_OWDC_SEC	PHOSPHORUS	PERIPHYTON
QW_RECMD_MTHDS	PHOS_SPECIES	MACROPHYTON
QW_TELEMETRY	NITROGEN	MICROINVERTS
QW_LST_UPDATE	N_SPECIES	MACROINVERTS
QW_PURPOSE	DETERGENTS	VERTEBRATES
QW_RECORDER_TYPE	OMI_CONSTITS	FUNGI
QW_RECORDER_FREQ	RADIOACTIVITY	VIRUSES
QW_PN_CODE	RCHM_SPECIES	BIO_RECMD_MTHDS
STORET_POINTER	CARBON	BIO_BEGIN_YR
QW_MODIFIERS	ORG_GROUPS	BIO_END_YR
KEY-LOC **	PEST_SPECIES	BIO_LST_UPDATE
	OTH_ORG_SPECIES	BIOLOGIC_MED
	BIOCHEM_OX_DMND	PRIMARY_PRDCTVTY
	CHEM_OX_DMND	SCENDARY_PRDCTV
	DISSOLVED_OX	CHEMOSYNTHETIC_A
	OTHER_DIS_GAS	BIOSTIMULATORY_T
	CHEM_RECMD_MTHDS	TOXICITY_TEST
	CHM_BEGIN_YR	OTHER_BIOASSAY_T
	CHM_END_YR	CHM_TISSUE_ANALY
	CHM_LST_UPDATE	HISTOPATH_ANALYS
	CHEMICAL_MED	OTHER_TISSUE_ANA
	CHM_MODIFIERS	BIO_MODIFIERS
	KEY-LOC **	KEY-LOC **

** Physical record number of corresponding record in LOC.PAT file.

TABLE C1. List of items in each file in INFO directory -- Continued

FILE NAME: PHY_DATA	FILE NAME: SED_DATA	FILE NAME: MET_DATA
NAWDEX#	NAWDEX#	NAWDEX#
TEMPERATURE	BED_LOAD	MET_BEGIN_YR
SPEC_CONDUCT	CNCNTRIN_SUS	MET_END_YR
TURBIDITY	CNCNTRIN_TOT	MET_INTERRUPTED
COLOR	PART_SIZ_SUS	MET_RAINFALL
ODOR	PART_SIZ_BED	MET_UNIT_RAINFAL
PH	SED_DIS_SUS	MET_AIR_TEMPERAT
SUSPD_SOLIDS	SED_DIS_TOT	MET_RSVD1
PHY_RECMD_MTHDS	SED_RECMD_MTHDS	MET_WIND_VELOCIT
PHY_BEGIN_YR	SED_BEGIN_YR	MET_RSVD2
PHY_END_YR	SED_END_YR	MET_RSVD3
PHY_LST_UPDATE	SED_LST_UPDATE	MET_RECMD_MTHDS
PHYSICAL_MED	SEDIMENT_MED	MET_OTHER
PHY_MODIFIER	SED_MODIFIERS	MET_TELEMETRY
KEY-LOC **	KEY-LOC **	MET_LST_UPDATE
		MET_MEDIA
		MET_RECORDER_TYP
		MET_RECORDER_FRE
		MET_PN_CODE
		MET_MODIFIERS
		KEY-LOC **

** Physical record number of corresponding record in LOC.PAT file.

REFERENCES

Lanfear, Kenneth J., and Hitt, Kerie J., 1988, Efficient operations on large geographic information system coverages: in Proceedings of the Eighth Annual ESRI User Conference, March 21-25, 1988, Palm Springs, California, [Redlands, California] Environmental Systems Research Institute, 12 p.

USING A GEOGRAPHIC INFORMATION SYSTEM TO DETERMINE PHYSICAL BASIN CHARACTERISTICS FOR USE IN FLOOD-FREQUENCY EQUATIONS

By

James J. Majure and P.J. Soenksen

ABSTRACT

A set of computer programs, named Basinsoft, was developed to use digital cartographic data to compute basin characteristics that are hypothesized to be related to floods. The programs work in conjunction with a proprietary geographic information system. Three digital cartographic data sets (coverages) of drainage basin boundaries, streams, and selected topographic contours are used by Basinsoft to calculate 16 basin characteristics: total drainage area, noncontributing drainage area, contributing drainage area, main stream length, basin length, channel slope, basin slope, basin width, shape factor, total stream length, drainage density, basin relief, ruggedness number, basin perimeter, relative relief, and drainage frequency. Basinsoft also plots the three coverages along with selected basin characteristics. The results produced by Basinsoft may be directly useful to hydrographic and geomorphologic studies, but its ultimate value will be to provide basin characteristics that can be used to improve flood-discharge and flood-frequency predictions.

INTRODUCTION

Determining the magnitude and frequency of floods at any site is an important step in the economical planning and safe design of bridges, culverts, levees, and retention structures, and is essential for the management of flood plains. At gaging stations where flood data have been collected for a number of years, these determinations can be easy to make. However, the great majority of basins in Iowa have no flood data, and methods that transfer data from other sites need to be used.

Previous investigations by the U.S. Geological Survey on the magnitude and frequency of floods in Iowa were prepared by Schwob (1953, 1966) and Lara (1973, 1987). The methods illustrated in these reports used data that were regionalized according to general basin types that were assumed to produce similar floods. Equations relating one or more measurable basin characteristic (drainage area, channel slope, mean annual precipitation) to the various flood-frequency characteristics were determined from available flood data within each region. Flood characteristics at ungaged sites were then estimated by using the appropriate regional equation and measured basin characteristics at the ungaged site.

The physical characteristics of a basin (size, ruggedness, drainage pattern, shape, soils, and so forth) largely control flood characteristics in that basin. However, limited cartographic data and the tedious procedures required to compute many physical characteristics of a basin, have prevented researchers from using all but a few of these characteristics in equations that estimate flood frequency. The composite effects of the remaining characteristics were dealt with in a general way through the use of large geographic regions with generally similar characteristics. However, there are significant limitations to this approach. Because the defined regions are not entirely homogeneous, and physical characteristics do not change abruptly at regional borders, individual basins, especially small basins, can be of a different type than that of the region they are in. In such cases the normal regional equations are not applicable, and there is no quantitative way of determining which, if any, of the other regional equations are applicable.

A more direct approach is to incorporate several basin characteristics into the equations so that flood characteristics can be directly related to quantitative measurements. Implicit to this approach is the capability to compute a variety of physical basin characteristics for a large number of basins where flood data have been collected, so the equations can be developed, and flood characteristics are desired. The advent of digitized cartographic data and geographic information systems (GIS) makes such large-scale computations not only possible, but also fast and accurate. This report presents the results of an investigation that used aGIS to determine physical basin characteristics for use in flood-frequency equations.

DETERMINING PHYSICAL BASIN CHARACTERISTICS USING A GEOGRAPHIC INFORMATION SYSTEM

Concepts

A set of computer programs, named Basinsoft, was developed to use digital cartographic data to compute basin characteristics that are hypothesized to be related to floods. This is accomplished by creating a representation of a drainage basin on a computer using the ARC/INFO1/geographic information system. Once the ARC/INFO representation of the drainage basin is entered into the computer, the information maintained by ARC/INFO and the computational capabilities it provides make it possible to calculate many basin characteristics.

The first step in creating a representation of a drainage basin is to define those characteristics of a basin that are to be determined. Because the number of basin characteristics that affect floods is large, it is not practical to determine all characteristics. For Basinsoft, the most important characteristics were chosen. These characteristics require that the contributing and noncontributing drainage areas, the streams, and the topography of a basin be represented. After determining which aspects of a basin are to be represented, how to best represent these aspects in the GIS needs to be determined. Geographic features are represented in ARC/INFO as polygons, lines, or points. A complete basin in Basinsoft is represented by three ARC/INFO coverages: a polygon coverage for the drainage area, a line coverage for the streams, and a line coverage for the topography.

The concept of having a complete basin in Basinsoft is important because when a command is given and the specified basin is not complete, Basinsoft will display an error message and abort. Specifically, a complete basin named <basin>, would consist of: a drainage area coverage, <basin>.BAS; a stream coverage, <basin>.STR; and a topography coverage, <basin>.CON. A complete basin also includes an INFO file named

<basin>.CHAR, that stores the basin characteristics calculated by the commands. If any of these four components of a Basinsoft basin are missing, the basin is not considered complete.

The User Interface

The Basinsoft commands are a set of ARC macros that work with short INFO programs to calculate basin characteristics from basin representations. The macros reside in a subdirectory of the ARC/INFO workspace in which the basin coverages are maintained. The INFO programs and the characteristics file reside in the workspace INFO directory.

To enter the system, execute the BASINSOFT.CPL file in the workspace. This CPL enters ARC/INFO, establishing the Basinsoft macros as system commands. It accepts, as an option, any legal command in ARC. A station file for the ARC/INFO session also may be specified after -STATION, -STAT or -ST. A station file identifies the hardware to be used by ARC/INFO. For example, the command: **OK, CPL BASINSOFT ARCPLOT -STATION IOWA**
would enter ARC/INFO, establish the Basinsoft macros as system commands, set the hardware characteristics to those specified in the IOWA station file, and enter the ARCPLOT subsystem.

The basic command format for the Basinsoft commands is:
<char> <basin> {recno} where:
<char> is the abbreviation of the characteristic to be calculated,
<basin> is the name of the basin, and
{recno} optionally specifies the record number in the characteristics file where the calculation will be stored.
For example, the command:
ARC: BS WILL 3
will calculate the basin slope (BS) for the basin named WILL, and will place the results of the calculation in the third record of the characteristics file. If a record number is specified, it needs to exist in the characteristics file.The {recno} option is included for flexibility. It allows multiple sets of characteristics to be maintained for a single basin. If omitted, the last record in the file will be updated. The contents of the updated record are overwritten.

Many of the characteristics calculated by Basinsoft use data that are automatically maintained by ARC/INFO, such as the area of polygons and the length of arcs. It is assumed that the coverage units are feet. The commands apply appropriate conversions to calculate the desired units.

The complete set of Basinsoft commands includes:

BN - basin name; a descriptive 1- to 40-character basin name. Entered after prompt.

TDA - total drainage area, in square miles, including noncontributing areas.

NCDA - drainage area, in square miles, that does not contribute to surface runoff at the basin outlet.

CDA - drainage area, in square miles, that contributes to surface runoff at the basin outlet.

BS - average basin slope, in feet per mile, which is computed as: BS = (length of all contours in miles) x (contour interval in feet) / CDA.

SL - stream length, in miles, is the length of the main stream from basin outlet to end of the defined channel.

BL - basin length, in miles, SL plus the extension of the defined channel to the basin divide.

CS - main channel slope, in feet per mile, is the difference in elevation at points 10- and 85-percent of BL divided by the distance between those points.

BW - effective basin width, in miles, which is computed as BW = CDA/BL.

SF - shape factor, dimensionless, is the ratio of effective basin width to basin length, which is computed as SF = BW/ BL.

TSL - total length of streams, in miles, which is computed by summing the lengths of all stream segments within the CDA.

DD - drainage density, in miles per square mile, which is computed as DD = TSL/ CDA within the CDA.

BR - basin relief, in feet, is the difference between the highest and lowest points within CDA. The user is prompted for these elevation values.

RN - ruggedness number, in feet per mile, which is computed as RN = TSL x BR/ CDA.

BP - basin perimeter, in miles, is the length of the entire basin divide.

RR - relative relief, in feet per mile, which is computed as RR = BR/BP.

DF - drainage frequency, in stream segments (arcs) per square mile, computed as DF = (number of stream segments)/CDA within CDA.

BASINC - calculates the entire set of basin characteristics for a basin.

MAPBASIN - creates a map of the basin and the following ten characteristics: TDA, CDA, BS, CS, SF, DD, BR, RN, RR, and DF. An example of MAPBASIN output is shown in figure 1.

ADDCHREC - adds a record to the characteristics file.

STARTBASIN - is a utility program to help set up a basin that is to be digitized from one 7.5-minute quadrangle. It prompts for the projection of the quadrangle and the smallest latitude and longitude values. It then creates empty basin coverages, with coverage units feet, and the characteristics file.

Coverage And File Structure

As explained above, each basin in Basinsoft is represented by three ARC/ INFO coverages. The three coverages need to be maintained in feet and named as follows: <basin>.BAS, <basin>.STR, and <basin>.CON. The characteristic file needs to be named <basin>.CHAR. In each case, <basin> is a 1-to 6-character basin name. For example, the basin named WILL consists of the three coverages, WILL.BAS, WILL.STR, and WILL.CON, and the characteristic file named WILL.CHAR.

Figure 1.--An example of MAPBASIN output.

Williams Cr nr Petersville, IA

TDA	=	1.795	mi^2	DD	=	2.18	mi/mi^2
CDA	=	1.795	mi^2	BR	=	145	ft
BS	=	380.40	ft/mi	RN	=	315.61	ft/mi
CS	=	0.00	ft/mi	RR	=	26.17	ft/mi
SF	=	0.593		DF	=	5.01	$streams/mi^2$

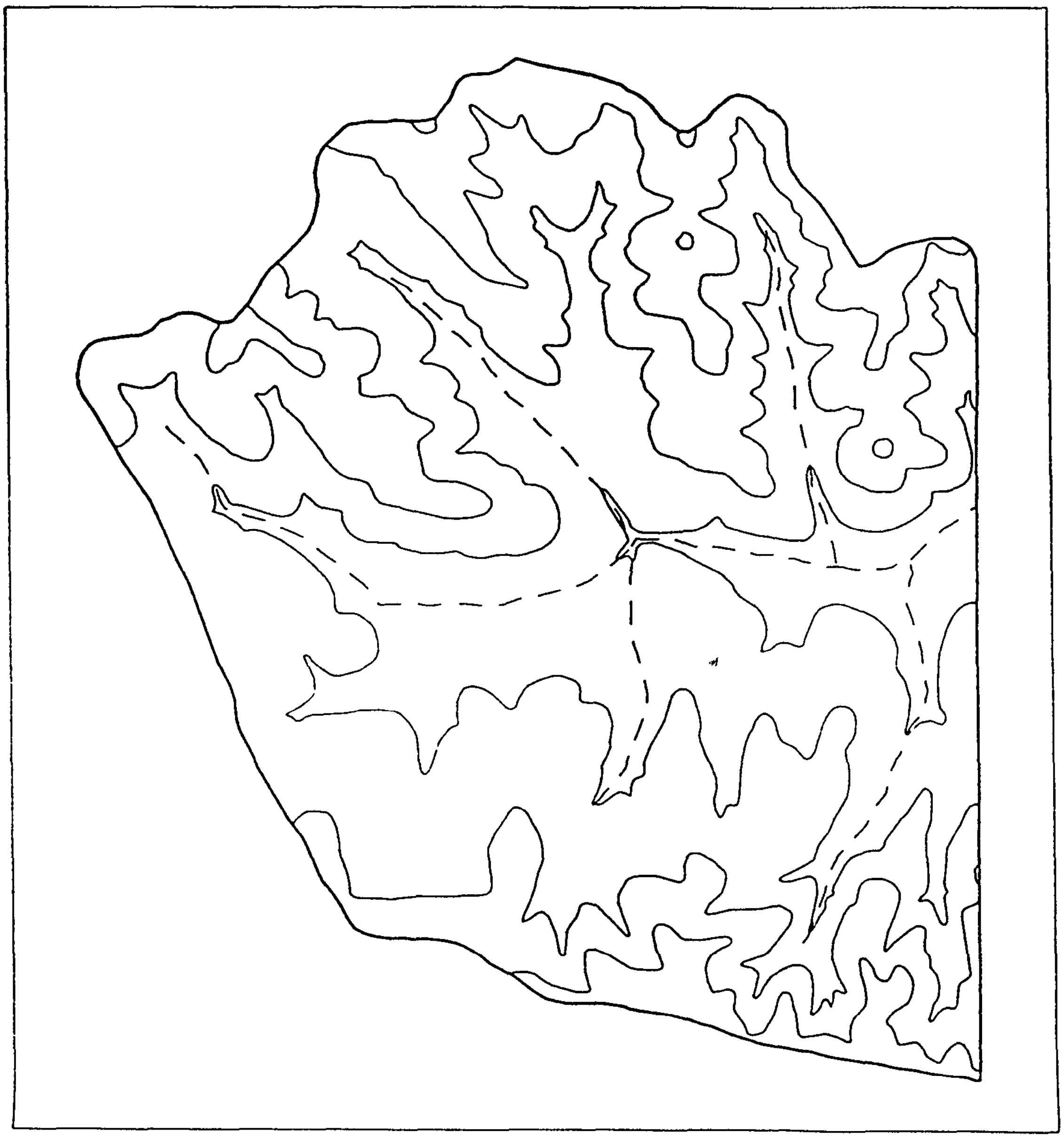


Figure 2.--File structure of the <basin>.BAS.PAT file

DATAFILE NAME: <basin>.BAS.PAT
5 ITEMS: STARTING IN POSITION 1

COL	ITEM NAME	WDTH	OPUT	TYP	N.DEC	ALTERNATE NAME
1	AREA	4	12	F	3	
5	PERIMETER	4	12	F	3	
9	<basin>.BAS#	4	5	B	-	
13	<basin>.BAS-ID	4	5	B	-	
17	CONTRIB	1	1	I	-	

The <basin>.BAS coverage is a polygon coverage that defines the basin area. For most basins, this coverage will contain one arc that defines the perimeter of the basin. For those basins that have noncontributing drainage areas, this coverage will contain additional polygons identifying the noncontributing areas. The item, CONTRIB, in the coverage PAT file will be used to determine whether a polygon represents a contributing or noncontributing drainage area. A value of 0 in CONTRIB indicates a noncontributing drainage area and a value of 1 indicates a contributing drainage area. The structure of the <basin>.BAS.PAT file is shown in figure 2.

The <basin>.STR coverage is a line coverage of all streams in the basin. The <basin>.STR.AAT file has the item CODE in addition to those maintained by ARC/INFO. This attribute can contain one of three values: 0, secondary stream; 1, main stream; and 2, extension line. All of the arcs that comprise the one main stream have a CODE value of 1. All other arcs that represent streams have a CODE value of 0. There is only one arc with a CODE value of 2. This arc is added to the end of the main stream and extends to the basin divide. This arc does not represent a stream, but is used to calculate the BL characteristic. The structure of the

Figure 3.--Structure of the <basin>.STR.AAT file

DATAFILE NAME: <basin>.STR.AAT
8 ITEMS:STARTING IN POSITION 1

COL	ITEM NAME	WDTH	OPUT	TYP	N.DEC	ALTERNATE NAME
1	FNODE#	4	5	B	-	
5	TNODE#	4	5	B	-	
9	LPOLY#	4	5	B	-	
13	RPOLY#	4	5	B	-	
17	LENGTH	4	12	F	3	
21	<basin>.STR#	4	5	B	-	
25	<basin>.STR-ID	4	5	B	-	
29	CODE	1	1	I	-	

Figure 4.--Initial structure of the <basin>.CON.AAT file.

DATAFILE NAME: <basin>.CON.AAT
8 ITEMS: STARTING IN POSITION 1

COL	ITEM NAME	WDTH	OPUT	TYP	N.DEC	ALTERNATE NAME
1	FNODE#	4	5	B	-	
5	TNODE#	4	5	B	-	
9	LPOLY#	4	5	B	-	
13	RPOLY#	4	5	B	-	
17	LENGTH	4	12	F	3	
21	<basin>.CON#	4	5	B	-	
25	<basin>.CON-ID	4	5	B	-	
29	ELEV	4	4	I	-	

<basin>.STR.AAT file is shown in figure 3.

The <basin>.CON coverage is a line coverage that contains a series of topographic contours for the basin. There can be as many contours as desired as long as the interval is constant. Initially, the <basin>.CON.AAT file has the item, ELEV, in addition to those maintained by ARC/INFO. ELEV will contain the elevation of each contour arc. The initial file structure of the <basin>.CON.AAT file is shown in figure 4.

Once the <basin>.BAS coverage and the <basin>.CON coverage have been created, coded, and cleaned up, the ARC/INFO IDENTITY command needs to be run on <basin>.CON, using <basin>.BAS as the identity coverage. The resulting coverage will contain all the attributes of both initial coverages. All items in the <basin>.CON.AAT file, except those maintained by ARC/INFO, ELEV, and CONTRIB must be deleted and the coverage renamed to <basin>.CON.This effectively allows contours that plot within

Figure 5.-- The final structure of the <basin>.CON.AAT file

DATAFILE NAME: <basin>.CON.AAT
9 ITEMS : STARTING IN POSITION 1

COL	ITEM NAME	WDTH	OPUT	TYP	N.DEC	ALTERNATE NAME
1	FNODE#	4	5	B	-	
5	TNODE#	4	5	B	-	
9	LPOLY#	4	5	B	-	
13	RPOLY#	4	5	B	-	
17	LENGTH	4	12	F	3	
21	<basin>.CON#	4	5	B	-	
25	<basin>.CON-ID	4	5	B	-	
29	ELEV	4	4	I	-	
33	CONTRIB	1	1	I	-	

noncontributing areas to be differentiated from those that do not. The final structure of the <basin>.CON.AAT file is shown in figure 5.

Arc Macro And INFO Program Structure

One of the most attractive features of this set of programs is its simplicity. Most of the information needed to calculate the basin characteristics is automatically maintained by ARC/INFO. Often, all that is needed are some simple calculations and unit conversions. Each command to calculate a basin characteristic consists of an ARC macro and a short INFO program. Almost all of the ARC macros are identical except that they initiate different INFO programs. The INFO programs are identical except for several lines that calculate the specific basin characteristic. Adding new basin characteristics can be done by adding an appropriate item to the characteristics file, and modifying an existing ARC macro and INFO program to calculate the new characteristic.

The ARC macro, TDA.AML, is shown in figure 6, and the INFO program, TDA.PG, is shown in figure 7. Comments are included indicating which lines are changed to create other commands.

Note that due to limitations in INFO one slightly unorthodox programming technique was needed in order to keep the programs generic or able to operate on a user specified basin and not a specific one.In these programs, INFO data files are SELECTed using the following two lines: **CONCATENATE $CHR2 FROM 'SEL ',$CHR1,'rest of file name' EXEC $CHR2** The SELECT command is created from the basin name stored in $CHR1, and is put in $CHR2. The command in $CHR2 is then executed by the EXEC command. Because the value of the variable executed by the EXEC command is not determined until run time, no file is explicitly selected during compilation and subsequent data item references will cause errors. To resolve this problem, duplicate INFO data files have been set up with the same file definitions as the corresponding actual files and are SELECTed prior to the EXEC command. Because a file with the proper structure has been SELECTed, compilation finishes without errors. A consequence of this method of selecting files is that anytime the file definition of an INFO file referenced by one of the programs changes, the definition of the corresponding duplicate file also needs to reflect the change and all programs that reference the file need to be recompiled. Otherwise, errors will occur, but no warning will be given!

Figure 6.--The ARC macro, TDA.AML

```
&args basin recno
&s basin [translate %basin%]
&if [null %basin%] &then
    &return &warning Usage:TDA
    <basin> {recno} /* This line
    changes...
&if ^ ( [exists %basin%.bas -coverage] &
    [exists %basin%.con -coverage] ~
    & [exists %basin%.str -coverage]
    & [exists %basin%.char -info] )
    &then &return &warning Basin,
    %basin%, is incomplete or does not
    exist.
&if [null %recno%] &then &s recno 0
&if ^ ( [type %recno%] = -1 ) &then
    &return &warning "recno" must be
    an integer record number.
&system como -ntty
down info
&data info
ARC
```

```
RUN TDA.PG /* and this line changes
%basin%^
%recno%
Q STOP
&end
up
&system como -tty
&return
```

Figure 7. --The INFO program, TDA.PG.

```
PROGRAM TDA.PG
FO $CHRl,8,8,C
FO $CHR2,80,80,C
FO $NUM21,4,12,F,3
FO $NUM23,4,4,I
ACC $CHR1      /* Receive the basin name
ACC $NUM23   /* Receive the char file
      recno
CMD COMO -TTY
SEL FAKE.CHAR ;use following EXEC to
      select real file;
CONCATENATE $CHR2 FROM 'SEL',
      $CHRl,'.CHAR'
EXEC $CHR2
IF $NOSEL EQ 0
      ADD TDA FROM ADD.DUMMY
ENDIF
      IF $NUM23 NE 0
      RES BY $RECNO = $NUM23
      IF $NOSEL EQ 0
         CONCATENATE $CHR2 FROM
         'Specified record doesn't exist in ',
         $CHRl,'.CHAR.'
         DIS $CHR2
         GOTO END
      ENDIF
ENDIF
SEL FAKE.BAS.PAT ;use following EXEC
      to select real file;   /*These
CONCATENATE $CHR2 FROM 'SEL
      ',$CHRl,'.BAS.PAT'/*lines
EXEC $CHR2               /*change
RES BY $RECNO  = 1    /*for each
CALC $NUM21 = 1 - AREA
                              /*characteristic.
SEL FAKE.CHAR ;use following EXEC to
      select real file;
CONCATENATE $CHR2 FROM 'SEL
      ',$CHRl,'.CHAR'
EXEC $CHR2
IF $NUM23 EQ 0
      RES BY $RECNO EQ $NOSEL
ELSE
      RES BY $RECNO EQ $NUM23
ENDIF
CALC TDA = $NUM21    .00000003587
      /* This line also changes.
LABEL END
MO -NTTY
END
```

<u>**Efforts To Automate Data Entry**</u>

Once the coverages representing a basin are in place, the Basinsoft commands are easy and straightforward. Digitizing the basins, however, can be a tedious and lengthly process.In order to decrease the time spent on this process, several possible ways of automating the data entry have been investigated.One possibility is acquiring digital line-graph data for hydrography and topography. This method still requires someone to define the drainage-basin perimeter and any noncontributing drainage area in the basin. Another possibility is acquiring digital elevation-model data. Programs exist that can determine the basin perimeter as well as any noncontributing drainage area, generate streams, and generate topographic contours from digital elevation-model data. This information can then be converted into ARC/INFO format and entered into the Basinsoft programs. The biggest problem with both of these possibilities is the major task of producing this digital data for all of the areas that need to be analyzed.

CONCLUSIONS

Basinsoft offers a means of relating flood-frequency predictions more directly to physical characteristics of individual drainage basins. Basinsoft provides a framework to which additional data describing the physical characteristics of a basin can be added as such data are obtained. This should ultimately help make predictions of flood frequency as accurate as possible.

REFERENCES

Lara, O.G., 1973, Floods in Iowa--Technical manual for estimating their magnitude and frequency: Iowa Natural Resources Council Bulletin 11, 56 p.

------1987, Method for estimating the magnitude and frequency of floods at ungaged sites on unregulated rural streams in Iowa: U.S. Geological Survey Water-Resources Investigations Report 87-4132, 34 p.

Schwob, N.H., 1953, Iowa floods, magnitude and frequency: Iowa Highway Research Board Bulletin I, 171 p.

------1966, Magnitude and frequency of Iowa Floods: Iowa Highway Research Board Bulletin 28, 45 p.

CHAPTER D--COMPUTER GENERATED GRAPHICS

THE INTEGRATION OF COMPUTER GRAPHICS AND TEXT-EDITING PROGRAMS

By Donald R. Block[1]

ABSTRACT

It is not possible to edit text parts of a graphics display while in CA-TELLAGRAF because it does not have an internal text editor. A FORTRAN-77 program called NCTELAGRAF was written to enable the user to retain graphics screen display during a CA-TELLAGRAF session while internally using PRIMOS command-file text editors without leaving CA-TELLAGRAF. This program allows use of text editors EM, ED, and WM and the RUN and SYS commands within CA-TELLAGRAF. The RUN command sends the edited file to CA-TELLAGRAF for execution. The SYS command invokes the PRIMOS emulator, which permits the user to enter commands directly to the operating system.

INTRODUCTION

Computer-generated graphics is rapidly becoming a standard method of producing graphs and charts. In the U.S. Geological Survey, computer-generated graphs and charts are used to display and help interpret hydrologic data. Computer software packages that enable the analyst to easily produce such graphs and charts are required tools in many hydrologic investigations. Many of the computer-generated graphics produced by Survey personnel are products of Computer Associates' CA-TELLAGRAF or CA-DISSPLA software run on PRIME minicomputers. CA-TELLAGRAF is primarily used for producing graphs from small data sets and for presentation graphics. CA-DISSPLA is a group of FORTRAN subroutines used by programmers to perform repetitive tasks requiring interaction of graphs with large data bases. CA-TELLAGRAF is relatively user friendly when compared to CA-DISSPLA.

CA-TELLAGRAF can be run in an interactive or batch mode. During an interactive session, CA-TELLAGRAF receives commands directly from the user, whereas during a batch session, it reads commands from a previously created command file. The most common methods of creating command files are: (1) using the "SAVE." command during an interactive CA-TELLAGRAF session, (2) entering CA-TELLAGRAF commands into a file using a text editor, and (3) running other software packages that create command files, such as Computer Associates' CueChart.

Advantages of using command files to run CA-TELLAGRAF are that (1) they can be easily modified to make changes in a graph and (2) they are often data-independent, allowing the production of the same style graph for different sets of data without modification. In the Survey's North Carolina, Nevada, and Tennessee District offices, libraries of command files are used by District personnel to produce graphs (R.C. Massingill, U.S. Geological Survey, written commun., 1988; J.C. Stone, U.S. Geological

---

[1]Computer Programmer Analyist, U.S. Geological Survey, 3916 Sunset Ridge Road, Raleigh, North Carolina 27607; telephone: (919) 571-4000.

Survey, oral commun., 1988). The disadvantage of using command files is that the exact command file required for a particular hydrologic application commonly is not available in the library, and modifying one of these command files using CA-TELLAGRAF is time consuming. Because CA-TELLAGRAF has no internal text editor, modifications are performed in an interactive manner. The user makes changes to a command file using an available external editor and then runs CA-TELLAGRAF to see the results of these changes. If further modifications are necessary, the user must repeatedly "quit" CA-TELLAGRAF, make the modifications, and then reinvoke CA-TELLAGRAF.

The time consumed alternating between CA-TELLAGRAF and external editors can frustrate the user. To alleviate this frustration, a FORTRAN-77 program called NCTELAGRAF[2] has been written to allow the use of Marc Software's WordMarc Composer+ and PRIME computer's EMACS and ED editors during an interactive CA-TELLAGRAF session. This paper describes the operation and programming of NCTELAGRAF.

NCTELAGRAF USER INFORMATION

NCTELAGRAF is invoked by typing "NCTELAGRAF filename". The filename argument refers to the user's CA-TELLAGRAF command file. If the filename argument is omitted, the program prompts the user to enter the name of his command file. Once the filename is entered, that file becomes the selected file inside NCTELAGRAF.

Once invoked, NCTELAGRAF submits the selected file to CA-TELLAGRAF for execution. After the selected file has been executed, control is returned to NCTELAGRAF. NCTELAGRAF looks and operates like CA-TELLAGRAF. Unlike CA-TELLAGRAF, NCTELAGRAF does not automatically clear graphics from the terminal screen after viewing. This allows the user to refer to the graphic while editing the command file. The user may clear the graphics area at any time using the terminal's graphics erase function. NCTELAGRAF also automatically appends a period to all user-entered commands before submitting them to CA-TELLAGRAF for execution, with the exception of the five NCTELAGRAF commands: EM, ED, WM, RUN, and SYS.

The EM, ED, and WM commands invoke the text editors made available by NCTELAGRAF. The EM command invokes the EMACS full screen editor. The ED command invokes the PRIMOS line editor. The WM command invokes the WordMarc Composer+ full screen editor. The RUN command submits the selected file to CA-TELLAGRAF for execution. The EM, ED, WM, and RUN commands can be entered with or without a filename argument. If the commands are entered with no filename argument, the selected file becomes the object of the command's action. If the filename argument is entered with the command, then that file becomes the selected file and is the object of the command's action.

The SYS command invokes NCTELAGRAF's "PRIMOS emulator". While in PRIMOS emulation, NCTELAGRAF imitates PRIMOS by allowing the user to submit commands directly to the operating system. To quit PRIMOS emulation and return to NCTELAGRAF, the user must type "Q". The user can also quit a NCTELAGRAF session and return to PRIMOS by typing "Q".

---

[2]All references to NCTELAGRAF refer to NCTELAGRAF version 2.0.

NCTELAGRAF PROGRAMMER INFORMATION

NCTELAGRAF is a Fortran-77 program that invokes CA-TELLAGRAF as a subroutine using the PRIME subroutine CP$. NCTELAGRAF submits all commands except EM, ED, WM, RUN, and SYS to CA-TELLAGRAF using a COMI (command input) file for an input buffer. This method has been described by Block and Gordon (1988, p. 100).

NCTELAGRAF submits command files to CA-TELLAGRAF as control files with the command "CONTROL FILE filename." where the filename argument is the pathname of the user's selected CA-TELLAGRAF command file. CA-TELLAGRAF is exited by default when a control file completes execution. To prevent this, NCTELAGRAF appends the line "RESET. CONTROL FILE KB." to all command files before submitting them to CA-TELLAGRAF for execution. Immediately after the control file has been executed by CA-TELLAGRAF, NCTELAGRAF takes the terminal out of graphics mode to prevent CA-TELLAGRAF from clearing the graphics area after the graphic is drawn. Prior to allowing user input, the command "CONTROL FILE 'DUMMY'." is submitted to CA-TELLAGRAF for execution. This command opens a control file named "DUMMY", which contains no information, but opening it causes CA-TELLAGRAF to close the user's selected command file. The selected command file can then be opened for editing by NCTELAGRAF.

The NCTELAGRAF commands EM, ED, WM, and SYS use the PRIME subroutine CP$ to submit commands to the PRIMOS operating system. When the process invoked by one of these commands completes execution, control is returned to NCTELAGRAF. The NCTELAGRAF RUN command checks the selected file for the line "RESET. CONTROL FILE KB.", appends this line if necessary, and then submits the commands "RESET. RESET. CONTROL FILE filename." to CA-TELLAGRAF. These commands instruct CA-TELLAGRAF to reset all variables to their default values before executing the control file. As before, after the control file has been executed, the command "CONTROL FILE 'DUMMY'." is submitted to CA-TELLAGRAF and results in the closing of the user's selected command file.

When the user types a "Q" in NCTELAGRAF, the command is submitted to CA-TELLAGRAF, ending the CA-TELLAGRAF session. This causes the CP$ subroutine that invoked CA-TELLAGRAF to return control to the main section of NCTELAGRAF allowing it to complete execution.

NCTELAGRAF SYSTEM INFORMATION

The NCTELAGRAF.F77 program is delivered with the accompanying files:

NCTELAGRAF.READ.ME.1ST - the installation instructions and user's manual,

NCTELAGRAF.INSTALL.CPL - a program to compile and bind NCTELAGRAF.F77, and

NCTELAGRAF.USERS - a list of people to whom the software has been sent.

The files occupy 14 PRIME records of disk space. The software can be installed for a single user or for multiple users. NCTELAGRAF requires the following conditions to operate correctly without modification:

1. The commands EM, ED, and WM invoke PRIME's EMACS, PRIME's ED, and Marc Software's WordMarc Composer+, respectively.

2. The file named TAGPRO.DAT exists in the directory from which NCTELAGRAF is invoked, and this file is correct for the terminal being used. Refer to the CA-TELLAGRAF user's manual for more instructions on setting up a TAGPRO.DAT file.

3. Every user has an active abbreviations file. This is required for NCTELAGRAF's PRIMOS emulator because NCTELAGRAF inserts "AB -EE" at the beginning of all commands submitted to PRIMOS. This allows the PRIMOS emulator to expand user abbreviations and will cause an error if no active abbreviations file exists.

NCTELAGRAF works with Revision 6.0 of CA-TELLAGRAF at Revision 2.0 of PRIMOS and is available upon request to the District Chief, U.S. Geological Survey, 3916 Sunset Ridge Road, Raleigh, North Carolina 27607. Installation instructions are included.

SUMMARY

Many computer-generated graphics produced by Survey personnel make use of CA-TELLAGRAF software for producing graphs and other presentation graphics. However, because CA-TELLAGRAF has no internal text editor, the editing of text within the graphics is a time-consuming iterative process whereby the user must exit CA-TELLAGRAF each time to make text modifications and then reenter the program to view the results of changes.

The NCTELAGRAF program essentially provides an internal text-editing capability during operation of the CA-TELLAGRAF program without exiting from CA-TELLAGRAF. Thus, on-screen text changes may be made and results immediately evaluated. The EM, ED, and WM commands are the text editors made available by NCTELAGRAF. Operating commands RUN and SYS, respectively, submit the edited file to CA-TELLAGRAF for execution and invokes NCTELAGRAF's PRIMOS emulator. NCTELAGRAF-user, programmer, and system details are provided.

REFERENCES

Block, D.R., and Gordon, J.A., 1988, COLLAGE: a menu-driven graphics editing software package, *in* Proceedings of computer associates graphics users' group meeting, Nashville, Tenn., Feb. 29-Mar. 1, 1988: p. 93-102.

Integrated Software Systems Corporation, 1985, TELL-A-GRAF user's manual, version 5.0: San Diego, Calif., Integrated Software Systems Corporation.

USE OF THE GRAPHICAL KERNEL SYSTEM STANDARD FOR HYDROLOGIC APPLICATIONS

By Thomas C. Wood and Alan M. Lumb

ABSTRACT

The Graphical Kernel System is an internationally accepted standard for computer graphics programming that allows for maximum portability of applications between different hardware platforms. Although the Graphical Kernel System standard is language-independent, it provides specific language bindings, such as Fortran, which are used for source code development. The Graphical Kernel System standard addresses the main functional areas of computer graphics programming, including output, coordinate systems, segmentation, interactive input, and "meta" file generation. Graphical Kernel System implementations are being developed on an increasing number of hardware platforms. The implementations most useful to the U.S. Geological Survey are those on Prime and UNIX-based workstations.

A set of Fortran utilities have been developed using the Graphical Kernel System for a range of surface-water programs used by the Geological Survey. Graphics include time plots at scales of minutes to years, x-y plots, and probability plots. Non-time axes can be arithmetic or logarithmic. Probability plots with Gaussian transformations can be fraction, percent, or recurrence interval. The utilities are stored in a Geological Survey software library available on most Prime systems. Implementations also have been made on a UNIX-based workstation and a personal computer.

INTRODUCTION

The Graphical Kernel System (GKS) is an internationally accepted standard for computer graphics programming that allows for maximum portability of applications between different hardware platforms. GKS was developed as a result of an increasing demand for computer graphics standards that began during the late 1960's, when new technology made computer graphics available to more people than ever before.

Graphics displays and plotters were developed and in use by the early 1950's. Color displays were in use by 1962. Most conventional graphics hardware had emerged by 1965. However, few people were involved with computer graphics. It is estimated that only 100 display systems were installed worldwide in 1965, at an average cost of $400,000 each (Hopgood and others, 1983, p. 2). Hardware vendors provided software that took advantage of their particular hardware capabilities. This led to very different approaches to writing graphics software.

Growing Demand for Standards

During the late 1960's, new technology produced interactive graphics devices at much lower costs, which made graphics available to a larger population. These new devices offered new capabilities, yet there was still no standard approach to writing software to address them. The lack of a common approach to developing graphics systems led to the formation of different schools of graphics systems design. The different systems were generally (1) device-dependent, (2) application-dependent, and (3) environment-dependent (Enderle and others, 1987, p. 54).

During the early 1970's, the cost of graphics devices continued to plummet and their use spread rapidly. Graphics systems developers were growing tired of re-writing their software with each hardware change. Also, people wanted to exchange programs with their colleagues without re-writing every routine at the local installation. Computer graphics seemed ready for an attempt at standardization.

Standardization Activities

During the middle 1970's, numerous standards organizations around the world began developing ideas for graphics standards. Major efforts were undertaken in Germany, the United States, and at the international level. Portability of application programs was the primary target of the standardization attempts. This would be achieved by standardizing the interface between the application program and a combined set of graphics functions. Three main strategies emerged for developing this standard interface (Enderle and others, 1987, p. 54):

(1) Design a new graphics language. It is very difficult to introduce a new programming language.

(2) Design a graphics extension of an existing high-level language. This is easier, however, the compiler would require modification to add new

features and overhead. It is difficult to persuade users of a language to accept new overhead.

(3) Design a subroutine package, callable from existing high-level languages. Defining graphics functions using parameter lists is not very appealing, however, this is the only strategy that could be implemented without harming other interests. Therefore, a subroutine package became the vehicle for implementation of the GKS and other proposed graphics standards.

In Germany, the Committee for Information Processing founded the subcommittee "Computer Graphics" in 1975. This 13 member group actually developed the initial GKS. The first draft of GKS was published in 1977.

In the United States, the Association for Computing Machinery's Special Interest Group on Graphics (ACM-SIGGRAPH) founded its own Graphics Standards Planning Committee in 1975. The Graphics Standards Planning Committee developed another subroutine package, known as the CORE system.

On the basis of increasing interest in graphics standards around the world, the International Organization for Standards formed a graphics committee in 1977, named WG2, with the goal of producing an international standard. In 1978, WG2 reviewed both the CORE and GKS proposals. In 1979, GKS was chosen as the starting point for an international standard. The International Organization for Standards managed the long GKS review and approval process.

Graphical Kernel System Becomes an International and American Standard

In August 1985, the International Organization for Standards published GKS as an international standard. Soon after, the American National Standards Institute adopted GKS as an American National Standard. In November 1986, American National Standard GKS was adopted as a Federal Information Processing Standard, for use in the Federal Government.

DEFINITION OF THE GRAPHICAL KERNEL SYSTEM

GKS is a hardware-independent, language-independent, device-independent, application interface for two-dimensional graphics output and interactive graphics input, providing maximum portability of applications. GKS contains all basic functions for interactive and non-interactive graphics on a wide range of graphics equipment.

Hardware Independence

GKS is implemented through a subroutine package, which is called from a high-level language, such as Fortran. Practically all hardware platforms support at least one high-level language. Therefore, GKS is hardware independent.

Language Independence

The GKS standard defines its capabilities through abstract subroutine names, such as "POLYLINE", "SET TEXT HEIGHT", and so forth. GKS implementors translate these abstract subroutine names into high-level language specific subroutine names, such as "GPL", "GSCHH", and so forth, in Fortran. GKS may be implemented in any high-level language that provides the subroutine construct. Therefore, GKS is language independent.

Device Independence

The GKS standard is at such a level of abstraction that device peculiarities are shielded from the application program(American National Standard for Information Systems, 1985, p. 12). An important concept that helps GKS be device independent is the logical workstation. The GKS interface provides uniform input and output functions that are defined on logical workstations. There exists an interface that assigns the abstract input and output functions defined on the logical workstations to specific device drivers.

Main Components of the Graphical Kernel System

GKS addresses the main functional areas of computer graphics programming. It specifies standard output primitives that are used to generate pictures. Three coordinate systems are provided, as well as transformations between each. GKS also provides segmentation and interactive input, which when combined together, add a new dimension to computer graphics systems. The logical workstation concept allows for device-independent graphics programming. Two metafile formats are supported for picture transfer.

Output Primitives

One of the basic functions of a graphics system is to generate pictures. In GKS, pictures are composed of output primitives (fig. D6). The main output primitives are:

(1) Polyline: a line connecting a list of specified positions,

(2) Polymarker: a symbol drawn at a specified position,

(3) Text: a character string drawn at a specified position,

(4) Fill Area: a polygon filled with a specified color or pattern.

Primitive attributes, such as polyline color, text height, and so forth, may be specified individually by the application program through attribute setting routines. GKS also provides attribute bundle indices and tables for each primitive type as an alternative. Each output device available to GKS has a set of primitive bundle tables associated with it. The application specifies a bundle index number for each primitive type and leaves the attribute selection to the bundle table (fig. D7). An example might be drawing a line on a graph. The application specifies "POLYLINE BUNDLE INDEX 2", followed by "DRAW POLYLINE." If a color device is specified as the workstation, the line is drawn in red on the basis of the attributes specified in the bundle table for that device. If a monochrome device is specified as the workstation, the line is drawn in a dashed pattern in white.

Coordinate Systems and Transformations

GKS defines three coordinate systems that are used to display and manipulate graphics on any device (fig. D8). The first of these is world coordinates, a cartesian coordinate system. The application defines world coordinates by specifying units of choice in the X and Y direction. GKS defines an intermediate coordinate system, normalized device coordinates with abstract values of 0 to 1 in both X and Y directions. The third coordinate system used by GKS is the device-dependent device coordinates, which GKS determines from the device being used. A normalization transformation is used to map world coordinate space into normalized device coordinate space. Normalized device coordinate space is then mapped to device coordinates space by a workstation transformation.

An example might be an application that produces a graph with units of 0 to 100 on both X and Y axes, which the programmer determines by the data being portrayed. The programmer defines world coordinates as 0 to 100 in both X and Y directions. The output device for the program might be any number of graphic devices with different sizes of output area. In order for the graph to be output correctly on any device available to GKS, the specified world coordinates need to be transformed to device coordinates, which varies depending on the device. This is done through the intermediate coordinate system, normalized device coordinates. First, the world coordinates ranges of 0 to 100 are transformed into an arbitrary normalized device coordinate space, which ranges from 0 to 1 in both X and Y directions. Then GKS can transform the arbitrary normalized device coordinates space into device coordinates of the specified device and the graph is drawn correctly.

Segmentation

As stated earlier, GKS pictures are composed of output primitives, such as lines, text, and so forth. Output primitives may be grouped together in parts and may be addressed and manipulated by the application as single units called segments. An example of creating a segment with GKS abstract routines is:

```
CREATE SEGMENT (1)
DRAW POLYLINE (5,X,Y)
DRAW TEXT (XPOS,YPOS,'SEGMENT 1')
CLOSE SEGMENT.
```

Segments may be manipulated by changing their transformation. GKS provides utilities that can be used by the application to move, scale, and rotate segments, through segment transformations. Segment attributes can also be changed by the application. These attributes include:

(1) Visibility: indicates whether or not a segment is being displayed on the display surface.

(2) Highlighting: a segment may be emphasized in a device-independent way by changing its appearance. An example is blinking or emboldening the primitives that make up the segment.

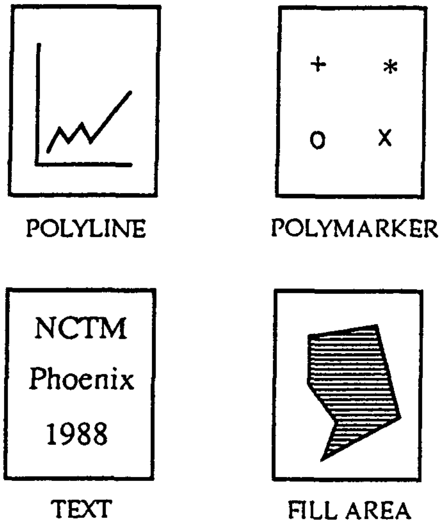


Figure D6.--Graphical Kernel System standard output primitives.

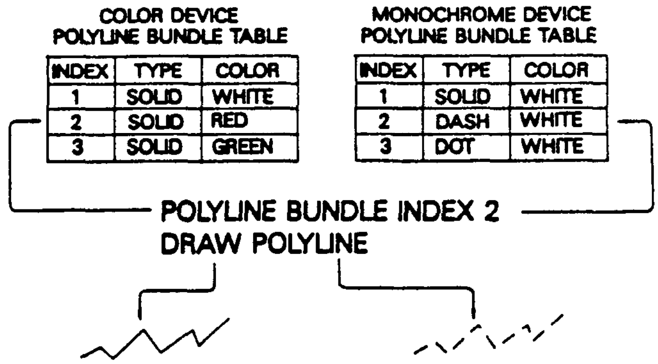


Figure D7.--Device independent output. Specifying "POLYLINE BUNDLE INDEX 2" produces a solid red line on color devices and a dashed line on monochrome devices, based on attributes set up in bundle table.

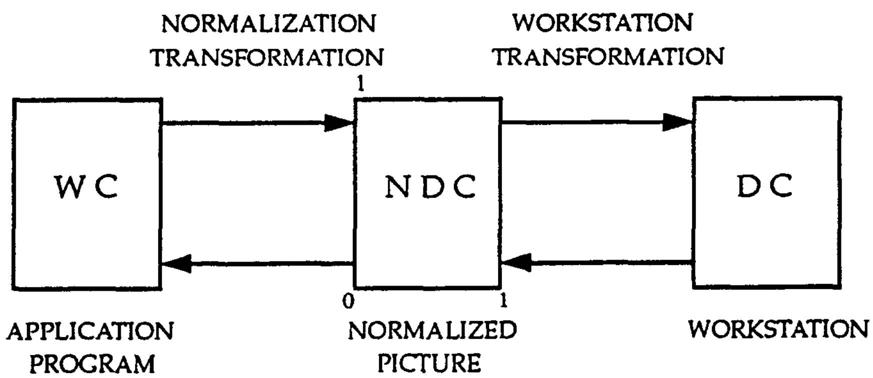


Figure D8.--Graphical Kernel System coordinate systems and transformations.

(3) Detectability: indicates whether or not a segment can be selected by the pick input function.

Segment manipulation works well on terminals that support segmentation in their hardware, such as Tektronix 4107 and higher models. GKS provides software emulation of segmentation on devices lacking segmentation, such as Tektronix 4010. However, segment manipulations performed through software are considerably slower than those managed by hardware.

Interactive Input

GKS provides a method for the application to request user input. This interactive input, combined with segment manipulation that is based on the input, adds a new dimension to computer graphics systems. A user of such a system is able to point to a certain part of the display (segment), move, erase, scale, rotate, and so forth, the selected part of the display.

GKS allows user input from several different logical input device classes, which are abstractions of physical devices; this provides device-independent input capability. Logical input devices are mapped to physical input devices, on the basis of a device driver selected by the application. Logical input classes include:

(1) Locator: provides a position in world coordinates, supplied by the user positioning a locator device. An example of a locator input device is a crosshair cursor moved around on a display by thumbwheels.

(2) Pick: provides a segment name, determined by the user positioning a locator on an output primitive contained in the segment.

(3) Valuator: provides a real number, supplied by the user, to be used for scaling, rotation, and other segment manipulation. Entering a number into a keyboard is the most common valuator input device.

(4) Choice: provides an integer which corresponds to the user's selection from a number of choices. These choices might be assigned to function keys on a keyboard, for example.

Logical Workstation Concept

GKS provides an application with device independence through the logical workstation concept. The logical workstation is an abstraction of the physical device. The application contains uniform input and output definitions that are mapped from the logical workstation to the actual device being used. A small section of program code defining the physical devices available to the application may be set up separately from the body of the application. The body of the application contains abstract references to input and output and is, therefore, completely portable. Input is referenced by logical input devices. Output is referenced by general drawing primitives and primitive attribute bundle tables (figs. D7 and D8).

GKS implementations provide some form of a workstation reference document, which details device-specific information about each device available to the particular GKS implementation. This information includes the bundle tables for each primitive type, input devices available for each logical input class, whether segmentation is available through hardware, and so forth.

Metafiles

Graphics metafiles provide a method of storing and retrieving pictures in an external file in a device and application-independent manner. GKS provides an interface to two metafile formats, GKS Metafile and Computer Graphics Metafile. Both metafile formats are available to GKS applications by specifying a corresponding workstation identifier. These identifiers are listed in the workstation reference document.

GKS Metafile is a format used only by GKS and is not part of the international standard, but is heavily used by GKS applications. A GKS Metafile records all output, including segment information, in a sequential format. GKS applications are able to read a GKS Metafile and display all or part of the stored picture, completely under user control.

The Computer Graphics Metafile is an American National Standards Institute standard and a draft international standard currently being processed by the International Organization for Standards. The Computer Graphics Metafile is being implemented by a rapidly increasing number of software vendors as an optional output format for graphics systems. A Computer Graphics Metafile stores a static picture and does not include segment information.

IMPLEMENTATION OF THE GRAPHICAL KERNEL SYSTEM

Practical implementation of GKS is done through language bindings that have been developed for several programming languages. There is a GKS implementation on most hardware platforms.

Language Bindings

GKS is defined in a language-independent manner, specifying abstract routines for graphical functions. These abstract routines are translated into language-specific routines for each particular programming language. This language-dependent layer is called a language binding. GKS has been translated into Fortran 77, PASCAL, C, and ADA programming languages. Each of these language bindings are also standardized in order to preserve portability of the applications. Standard routine names are derived from abbreviations for each corresponding GKS function. In the Fortran language binding, all routines begin with the letter "G" and end with up to five letters as an abbreviation for the function of the routine. For example, the GKS input function "REQUEST LOCATOR" is translated to routine "GRQLC".

Hardware Platforms

GKS has been implemented by numerous hardware and software vendors on most platforms, from microcomputers to supercomputers. Purchase prices for GKS implementations range from $500 to greater than $30,000. GKS implementations of most interest to the U.S. Geological Survey are those available for Prime systems and UNIX-based workstations.

The Geological Survey currently has a GKS implementation on every Prime in the Distributed Information System (DIS) network. Computer Associates' CA-GKS, using the Fortran binding, was introduced as part of the DISSPLA revision 10.0 upgrade in 1985. CA-GKS, although a separate package from DISSPLA, does share files with DISSPLA, including device drivers and metafiles. Other companies that offer GKS packages on the Prime include NOVA*GKS by Nova Graphics International, D-PICT/GKS by Pansophic Systems, and GRAFPAK-GKS by Advanced Technology Center.

The graphics requirements of DIS-II, the Geological Survey's next generation of computing tools, includes a GKS implementation. The main component of DIS-II will be a UNIX-based workstation. There is a rapidly increasing number of vendors offering GKS implementations on UNIX-based workstations. These include all of the packages mentioned above for Prime systems, as well as GK-2000 by Precision Visuals, Inc., and Visual: GKS by Visual Engineering.

WHEN TO USE THE GRAPHICAL KERNEL SYSTEM

When designing a graphics application, the existence of one or more of the following situations warrants consideration of using GKS for the application development (Federal Information Processing Standard, 1986, p. 2):

(1) The application will be programmed centrally for a decentralized system that might consist of different types of computers and graphics devices.

(2) It is probable that the life of the application will be longer than the life of the present graphics equipment.

(3) The application is likely to be used by organizations outside the Federal Government, such as state and local governments, universities, and so forth.

(4) The application will be run on equipment other than that on which it is developed.

(5) The application will be maintained by programmers other than the original ones.

Advantages of Developing an Application Using a Graphical Kernel System Implementation

(1) Maximum portability of applications between different hardware platforms.

(2) Limited dependence on a specific software vendor. GKS implementations may be purchased from numerous vendors.

(3) Access to new devices with minimal program modification, due to the logical workstation concept.

(4) Program development in commonly used languages, such as Fortran.

Disadvantages of Developing an Application Using a Graphical Kernel System Implementation

(1) More programming required to produce output than in most proprietary graphics packages. For example, drawing a chart axis requires individual calls to primitive drawing routines to draw the axis line, each tick mark, and the tick labels. A GKS program might require ten or more lines of code to draw an axis. A package like DISSPLA requires only one line of code to produce the entire axis.

(2) Additional code is required to translate logical workstation commands to those of a particular device. This additional overhead will sometimes slow a GKS application, compared to more device-specific software.

APPLICATION OF THE GRAPHICAL KERNEL SYSTEM FOR SURFACE WATER PROGRAMS

For all five of the reasons listed in the previous section, GKS was selected to develop graphics output for surface-water programs. These programs include several rainfall-runoff and flow-routing models as well as programs for flow-duration analysis, frequency analysis and generalized leasts squares. These programs are used throughout the Geological Survey on Prime's, UNIX-based workstations, and personal computers. Thus, portability is of primary interest and can provide major savings in the resources required to develop and maintain the necessary graphics software.

As stated earlier, one of the major disadvantages of GKS is the additional programming effort required. Thus, one goal for the surface-water graphics software was the development of a set of graphics subroutines that would reduce the level of programming for subsequent applications. The use of a common set of subroutines also reduces the effort required to maintain the software. A brief description of the subroutines and graphics products is contained in the following sections. These subroutines are a first effort at establishing a standard set of graphics subroutines. When sufficient experience has been gained, the subroutines will be refined, redesigned, or discarded. The goal is to provide several layers of graphics subroutines for use in surface-water programs. At the highest layer would be several subroutines with a minimum number of arguments. At the lowest layer would be the GKS subroutines. Layers in between would include Fortran subroutines that perform part of the graphic display such as a log axis, time axis, or explanation text. At the top levels, many items are defaulted. At the lower layers, the programmer has to set more items or request them from the user. When a change is made to another graphics standard, in theory, only the lower layer routines need to be changed.

Graphical Kernel System Subroutines

Table D1 lists the GKS subroutines that are used. A basic implementation of GKS is used with no segmentation or bundling. Only one workstation is open at a time.

Graphics Subroutines Usage

Application programmers can use the graphics subroutines without learning the details of GKS. Programming with these subroutines is similar to using DISSPLA or PLOT10. All the subroutines are in a library that may be obtained from the U.S. Geological Survey, Office of Surface Water, 415 National Center, Reston, Virginia 22092.

The subroutines were designed around two common blocks, one for the data to be plotted and one for all the text and specifications for the plot (table D2). When all the appropriate variables in the common blocks have been set, a routine is called to make the plot.

The graphics subroutines can be used with one of three approaches:

(1) linking selected subroutines in the users' program,

(2) filling the common block using a subset of the subroutines and linking the subroutine library, or

(3) partially filling the common block using selected subroutines, then letting the user interactively set or change variables with an additional set of subroutines.

The third approach uses the most subroutines from the library. All the subroutines are defined in a system documentation file also available from the Office of Surface Water. That document is computer generated to help guarantee accuracy and more easily generate current documentation. For each subroutine, the document includes name, type, and purpose. Each argument is defined, the type is specified, and each is classified as input, modify, or output. Common blocks and variables that are used are also listed.

The variables used in the common blocks are listed and defined in table D2. Note, the maximum number of values to be plotted is 6,000 and the maximum number of variables is 18. Each of those can be changed in a PARAMETER statement in the include file for common blocks.

Approach 1 -- A few of the subroutines do not use the common block, they scale the axes, physically size the axes and draw the axes. A list of these subroutines is found in table D3. Arguments for these routines are described in the system documentation. As many as 80 characters can be used in the axes titles.

Approach 2 -- If the subroutine library is used, the common blocks cannot be included in the users' software. For this case, the set of utilities listed in table D4 are used to pass the data to the common blocks. Then utilities GPSTUP and PLTONE are called to make the plot.

Table D1.--Graphical Kernel System subroutines used in utilities

FORTRAN 77 Name	GKS Function
GOPKS	OPEN GKS
GCLKS	CLOSE GKS
GOPWK	OPEN WORKSTATION
GCLWK	CLOSE WORKSTATION
GACWK	ACTIVATE WORKSTATION
GDAWK	DEACTIVATE WORKSTATION
GSWN	SET WINDOW
GSVP	SET VIEWPORT
GSWKWN	SET WORKSTATION WINDOW
GSWKVP	SET WORKSTATION VIEWPORT
GTX	TEXT
GFA	FILL AREA
GPL	POLYLINE
GPM	POLYMARKER
GSELNT	SELECT NORMALIZATION TRANSFORMATION
GQOPS	INQUIRE OPERATING STATE VALUE
GQDSP	INQUIRE DISPLAY SPACE SIZE
GQCF	INQUIRE COLOR FACILITIES
GQTXFP	INQUIRE TEXT FONT AND PRECISION
GQCHXP	INQUIRE CHARACTER EXPANSION FACTOR
GQCHSP	INQUIRE CHARACTER SPACING
GQCHH	INQUIRE CHARACTER HEIGHT
GQCHB	INQUIRE CHARACTER BASE VECTOR
GQTXX	INQUIRE TEXT EXTENT
GSASF	SET ASPECT SOURCE FLAGS
GSTXFP	SET TEXT FONT AND PRECISION
GSCHXP	SET CHARACTER EXPANSION FACTOR
GSCHSP	SET CHARACTER SPACING
GSTXAL	SET TEXT ALIGNMENT
GSCHH	SET CHARACTER HEIGHT
GSCHU	SET CHARACTER UP VECTOR
GSTXCI	SET TEXT COLOR INDEX
GSTXP	SET TEXT PATH
GSFACI	SET FILL AREA COLOR INDEX
GSFAIS	SET FILL AREA INTERIOR STYLE
GSFASI	SET FILL AREA STYLE INDEX
GSLN	SET LINETYPE
GSPLCI	SET POLYLINE COLOR INDEX
GSMKSC	SET MARKER SIZE SCALE FACTOR
GSPMCI	SET POLYMARKER COLOR INDEX
GSMK	SET MARKER TYPE

Table D2.--Variables in common blocks

COMMON block CPLOTB.INC

YX(n)	-values of data to be plotted (1 < n<6000)
BUFPOS(j,k)	-j =1,6 k=1,18
	-for time-series start (j=1)/end(j=2) positions in YX array for each time-series or variable plotted (positions 3,4 not needed).
	for x-y plots positions (j=1) and (j=2) for Y axis, positions (j=3,j=4) for X-axis
	for x-y plots positions (j=5,j=6) for size of symbol if CTYPE = 7

COMMON block CPLOT.INC

DEVCOD	-device code, system dependent number
DEVTYP	-output device category
	1-screen,2-printer(impact and laser),3-plotter
	4-GKS meta file, 5-DISSPLA meta file
FE	-Fortran unit number for GKS error file
NCRV	-number of curves
NVAR	-number of variables
PLMX(1)	-maximum value for Y-axis.
PLMN(1)	-minimum value for Y-axis.
PLMN(2)	-maximum value for Y-axis on right side
PLMN(2)	-minimum value for Y-axis on right side
PLMX(3)	-maximum value for auxilary axis.
PLMN(3)	-minimum value for auxilary axis.
PLMN(4)	-maximum value for X-axis
PLMN(4)	-minimum value for X-axis
YMIN(k)	-k=1,18 minimum value for each variable
YMAX(k)	-k=1,18 maximum value for each variable
TICS(k)	- number of tics on axes (default=10 except auxilary axis=2)
	k=1 for Y-axis on left
	k=2 for Y-axis on right
	k=3 for auxilary axis
	k=4 for X-axis
XTYPE	-type of X-axis
	0-time
	1-arithmetic
	2-logarithmetic
	3-probability percent (normal distribution) 99-1
	4-recurrence interval (normal distribution) 1-100
	5-probability fraction (normal distribution) .99-.01
	6-probability percent (normal distribution) 1-99
	7-recurrence interval (normal distribution) 100-1
	8-probability fraction (normal distribution) 0.01-.99
YTYPE(i)	-type of Y axis (i=1 for left axis) (i=2 for right axis)
	0-none (applies only to right axis, left axis must be non-zero)

```
                 1-arithmetic
                 2-logarithmetic
WHICH(k)        -which axis for each variable (k=1,NVAR)
                 1-left       y-axis
                 2-right      y-axis
                 3-auxilary
                 4-x-axis
CTYPE(k)        -type of curve (k=1,NCRV)
                 1-uniform time step with lines or symbols
                   (main plot)
                 2-uniform time-step with bars (main plot)
                 3-uniform time-step with lines or symbols
                    (auxilary plot on top)
                 4-uniform time-step with bars (auxilary plot on top)
                 5-non-uniform (date-tagged) time-series
                 6-x-y plot
                 7-X-Y plot with symbol sized on a third variable
DTYPE           -data type for time-series
                  0-mean or sum over time-step
                  1-instantaneous or point data
TITL(24)        -title for the plot.
YLABL(80)       -label for the y-axis.
YXLABL(80)      -label for other axis (XTYPE=2 or YTYPE(2) =0)
YALABL(80)      -label for auxilary plot on top
LBC(j,k)        -j=1,18 k=1,NCRV
                -label for the k-th curve (20 characters)
LBV(j,k)        -j=1, 18, k=1,NVAR
                -label for the k-th variable (20 characters)
TSTEP(k)        -time step for each curve in TUNITS (k), k=1,NCRV
TUNITS(k)       -time units for each curve (2-min,4-day,5-mo,6-yr) k=1,NCRV
SDATIM(6)       -starting year,month,day,hour,minute, second of plot.
EDATIM(6)       -ending year,month,day, hour,minute, second of plot.
SYMBL(k)        -code for symbol type (k=1,NCRV)
                SYMBOL                      GKS CODE
                ------------                ---------------
                NONE                        0
                .                           1
                +                           2
                *                           3
                O                           4
                X                           5

LNTYP(k)        -code for type of line (k=1,NCRV)
                LINE                        GKS CODE
                -------                     ---------------
                NONE                        0
                SOLID                       1
                DASH                        2
                DOT                         3
                DOT-DASH                    4
```

PATTRN(k) -Code for shading (k=1,NCRV)

PATTERN	GKS CODE
NONE	1
SOLID	2
HORIZ	3
VERT	4
DIAGONAL	5

COLOR(k) -code for color (k=1,NCRV)

COLOR	GKS CODE
background	0
B/W	1
RED	2
GREEN	3
BLUE	4
CYAN	5
MAGENTA	6
YELLOW	7

BCOLOR	-background color code number (0=white, 1=black)
YLEN	-length of y-axis in world coordinates (WC) or both main and auxilary axis plus small space between them.
XLEN	-length of x-axis (WC)
ALEN	-auxilary plot axis length (WC)
YPAGE	-vertical page size (WC).
XPAGE	-horizontal page size (WC).
XPHYS	-physical origin (WC) in horizontal
YPHYS	-physical origin (WC) in vertical
SIZEL	-height of lettering (WC)
LOCLGD(j)	-action for legend (j=1,2) -2.0=no legend -1.0=legend in upper left corner (default location) x,y=legend at fraction x and y of XLEN and YLEN from origin. (values between 0.0 and 1.0)
TRANSF(j,k)	-transformation type for each variable (k=1,NVAR) (j=1 for left main y-axis) (j=2 for right y-axis or x-axis) 0=none 1=arithmetic (no transformation) 2=logarithmic 3=Normal distribution
BVALFG(4)	-bad value flag for bottom,top,left,right 1=clip,plot at point going off scale 2=ignore, leave blank 3=plot arrow pointing off scale, don't connect lines 4=ignore,connect good points
BLNKIT(4)	-min-max on y-axis and min-max on x-axis for box for no plotting (fractions from 0.0 to 1.0)
CTXT(i)	-text to be placed on the plot (max 120 characters)
CPR	-characters per line

NCHR	-number of characters to use (up to 120)
FYT	-fraction (0.0-1.0) of YLEN for upper left corner of text
FXT	-fraction (0.0-1.0) of XLEN for upper left corner of text

Table D3.--Subroutines not requiring common blocks

AXAXIS -	draws arithmetic X-axis
AYAXIS -	draws arithmetic Y-axis
TMAXIS -	draws time axis based on start/end dates
PBAXIS -	draws probability X-axis in fraction, percent or recurrence interval
LXAXIS -	draws logarithmic X-axis
LYAXIS -	draws logarithmic Y-axis
SCALIT -	scales axes
SIZAXE -	sets origin, axes lengths and letter size based on size of plotting surface

Table D4.--Subroutines to fill common block

GPINIT -	initializes both common blocks
GPNCRV -	sets number of variables and curves
GPDATR -	adds data for a variable
GPTIME -	sets starting and ending time and time step and time units for each variable
GPLABL -	sets types of axes and titles
GPCURV -	sets symbol, color, line type, pattern and label for each curve
GPLEDG -	sets upper left corner for legend
GPTEXT -	sets supplementary text information to be printed on the plot
GPDEVC -	sets code for device type
GPSIZE -	set plot space, location of origin, axes lengths, and letter size
GPBLNK -	sets window for no graphics
GPVAR -	sets minimum and maximum for each variable and which axes for each variable
GPSCLE -	sets axes scales, number of tics, and action to be taken when plotting off scale
GPGNRL -	sets color of background
	GPERFL - sets Fortran unit number for GKS error file

Approach 3 -- At this level, the applications programmer can use additional utilities in the library. This will let the user interactively modify variables pre-selected by the programmer such as symbol, line type, color, axes titles, or axes scales. Approach 3 is similar to Approach 2 except a main program shell needs to be modified and used to initialize and set up the system. At this level, a user configuration file TERM.DAT, can be used to set device-dependent attributes such as font, text precision, symbols, line types, fill patterns, and colors. After some of the variables in the common blocks have been set, the programmer calls the subroutine PROPLT which has arguments to indicate which of the following groups of variables (1) can not be changed, (2) can be changed and (3) need to be changed:

- data
- device
- axes
- curve specifications
- titles
- extras
- scales
- sizes.

At this point, control is turned over to the user to plot, modify, and re-plot until the user is finished and control is passed back to the program that called PROPLT.

Use of the Program ANNIE for Graphics

Each of the types of plots can be created using the interactive program ANNIE (Lumb, and others, 1990). ANNIE is a computer program for interactive hydrologic analysis and data management. Data can be entered from one of three types of files, Watershed Data Management (WDM) file, PLTGEN file and flat file. Data also can be entered from the terminal. The WDM file is a binary, direct access file designed for storage and retrieval of data for surface-water application programs. The PLTGEN file is a sequential, formatted file for time-series data. The flat file or terminal input lets the user specify the starting position and length of each variable on the records in the file. Alternately, the file or terminal input can contain all the values for the first variable in free field format followed by all the values for the next variable in free field format, and so forth. For free field format, the values need to be separated by blanks or commas. The values can be on one or more records with a maximum record length of 132 bytes. Each variable in the free field format must start a new record. All of the other specifications for the plot are provided by way of menus.

Examples of ANNIE graphics are shown in figures D9 and D10. The first is a probability plot that illustrates the use of additional text. The second is a time-series plot of streamflow, evaporation, and precipitation for the 1957 water year. Precipitation is plotted on an auxiliary plot that is an option with time-series plots. The axis on the auxiliary plot is limited to arithmetic. The curve for time-series data was plotted as a step function for mean values, and was plotted as a line connecting points for instantaneous values.

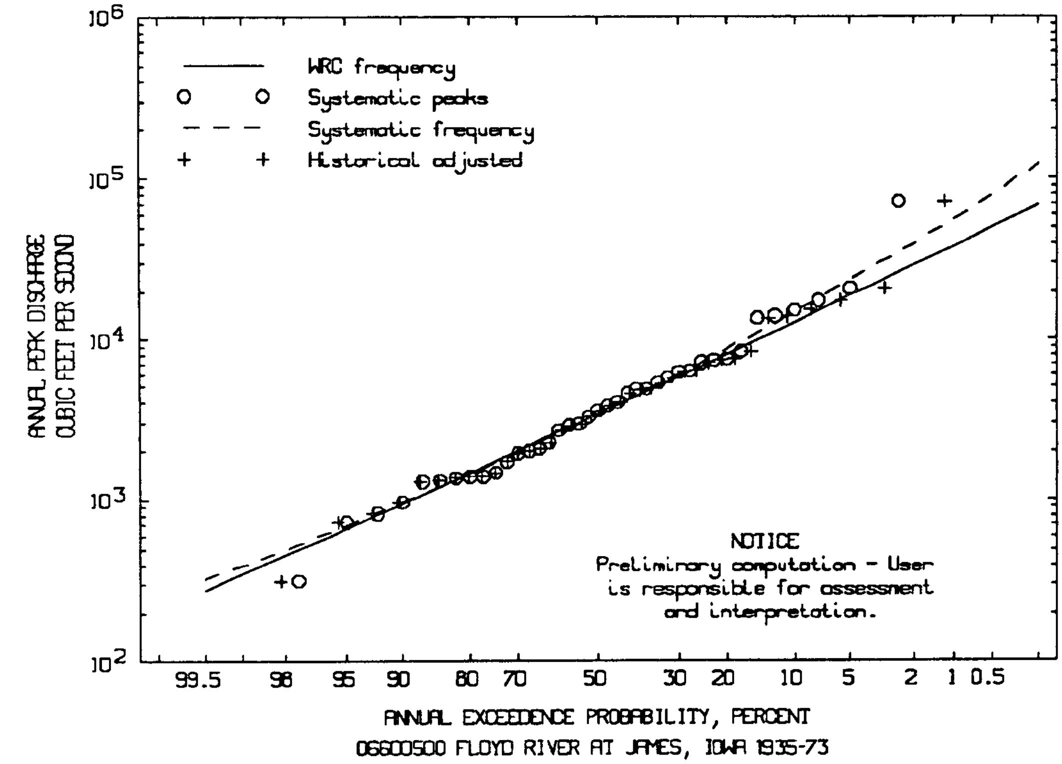


Figure D9.--Example of probability plot.

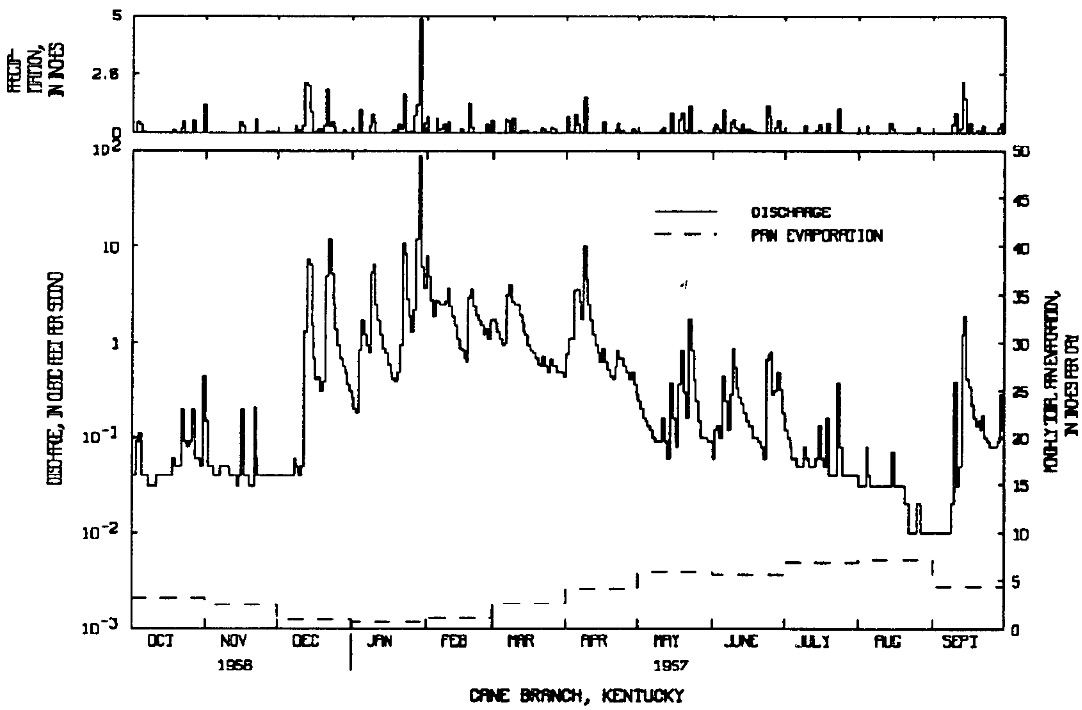


Figure D10.--Example of time-series plot.

EXPERIENCE WITH THE GRAPHICAL KERNEL SYSTEM

GKS was not easy to learn. Although training courses are available from vendors, none were used. Two textbooks, one by Hopgood, 1983, and the other by Enderle, 1987, were quite helpful.

The Fortran binding for GKS is at a basic level. Fortunately, the previous graphics subroutine library for ANNIE was implemented at a basic level in an attempt to meet publication standards. Thus, the translation was fairly straightforward for drawing lines, axes, and writing text. Most of the difficulty was experienced in learning how to use the transformation routines from the world coordinates, to normalized device coordinates, to device coordinates. Locating text on the plot was initially confusing. Additional text problems centered around implementation of GKS.

Currently, the biggest disadvantage of GKS on the Prime is the inadequate implementation that is used. Other vendors are being evaluated as a potential source. Implementation of GKS on the personal computer is quite good and one vendor has drivers for over 120 devices. The Geological Survey has purchased the development library and a run-time distribution license for personal computers.

SUMMARY AND CONCLUSIONS

During the 1970's, a growing demand for computer graphics programming standards emerged. Numerous standards organizations around the world began developing ideas for graphics standards. During the mid 1980's, the Graphical Kernel System was adopted as an international, American National, and Federal Government standard for computer-graphics programming.

The Graphical Kernel System is a hardware-independent, language-independent device-independent application interface for two-dimensional output and interactive input, providing maximum portability of applications. The Graphical Kernel System addresses the main functional areas of computer graphics programming. The main components of the Graphical Kernel System are output primitives, coordinate systems, segmentation, interactive input, logical workstation, and metafile generation.

The Graphical Kernel System is defined in a language-independent manner that specifies abstract routines. These abstract routines have been translated into standard language specific routines known as language bindings. The Graphical Kernel System has been implemented by numerous hardware and software vendors on most platforms, from microcomputers to supercomputers. The implementations most useful to the U.S. Geological Survey are those on Prime and UNIX-based workstations.

The Graphical Kernel System has been used to implement graphics software for surface-water applications. The disadvantages are the inadequate implementations by vendors and the limited number of device drivers. These disadvantages will be reduced as vendors continue to improve their implementations. The Graphical Kernel System was difficult to learn, but other developers in the Geological Survey can use the utilities instead of the Graphical Kernel System directly.

The disadvantages are outweighted by the major advantage of portability. With the expense of developing and maintaining software, a common software package for Prime's, UNIX-based workstations, and personal computers provides a major savings in resources.

REFERENCES CITED

American National Standard for Information Systems, 1985, ANSI X3.124 Computer Graphics-Graphical Kernel System (GKS) Functional Description, 268 p.

Enderle, G., Kansy, K., and Pfaff, G., 1987, Computer Graphics Programming. GKS - The Graphics Standard: New York, New York, Springer-Verlag, 632 p.

Federal Information Processing Standard, 1986, FIPS PUB 120 Graphical Kernel System (GKS), 3 p.

Hopgood, F. R. A., Duce, D. A., Gallop, J. R., and Sutcliffe, D. C., 1983, Introduction to the Graphical Kernel System: Orlando, Florida, Academic Press, Inc., 189 p.

Lumb, Alan M., Kittle, John L., Jr., and Flynn, Kathleen M., 1990, Users Manual for ANNIE, A Computer Program for Interactive Hydrologic Analysis and Data Management: WRI 89-4080, U.S. Geological Survey, 236 p.

LINKING DIGITAL TECHNOLOGY TO PRINTING TECHNOLOGY FOR PRODUCING PUBLICATION-QUALITY COLOR GRAPHICS

By Gregory J. Allord and Kerie J. Hitt

ABSTRACT

The U.S. Geological Survey has published thousands of high-resolution computer-generated graphs and maps in the Water-Supply Paper report series for the National Water Summary. The production of publication-quality computer graphics is accomplished by linking existing digital technology to pre-press technology. For example, Prime minicomputers running CA-TELLAGRAF and ARC/INFO software can be linked to output devices such as the Scitex and the Linotype L300. The hardware, software, and techniques used to prepare illustrations have evolved to keep pace with the increasing quantity and high complexity of the graphics required in each subsequent water summary. Examples of design criteria, choice of hardware and software, and methods of production used to prepare each report were chosen to illustrate this evolution. The goal has been to design computer data bases that are conducive to analysis and to effective displaying of data.

GRAPHICS ON MICROCOMPUTERS

By

R.T. Hanson

U.S. Geological Survey

Tucson, Arizona

ABSTRACT

The use of the graphics package PC-DISSPLA[1] has extended some of the capabilities of computer-graphics programs from mainframe computers to microcomputers. PC-DISSPLA is based on the VAX version of DISSPLA and has advantages and limitations with respect to the mainframe version. This package is used in the Arizona District of the U.S. Geological Survey for preparation of x-y plots, probability plots, text slides, and contour maps. A major advantage over the mainframe time-sharing environment is the ability to have redundancy and stand-alone graphics independent of mainframe processes, which can degrade the performance of graphics programs. Another advantage of processing on a microcomputer is the ability to transfer metafiles to and from mainframe computers; this takes advantage of hardware and software capabilities on both machines for post-processing tasks. PC-DISSPLA used in combination with the Graphic Software Systems peripheral-device library provides a wide variety of graphic devices including raster devices, plotters, and the post-script interface. Future versions of PC-DISSPLA will take advantage of enhancements in DISSPLA Revision 11.0 and the new Microsoft OS-2 environment. The major limitation with PC-DISSPLA is the size of programs that can be run on microcomputers in the DOS environment.

APPLICATION OF USER-SUPPLIED TRANSFORMATIONS IN COMPUTER-GRAPHICS PROGRAMS

By

Stanley A. Leake

U.S. Geological Survey

Tucson, Arizona

ABSTRACT

User-supplied transformations of plot vectors are a feature of the CA-DISSPLA software package. The transformations, which involve changing the coordinates of the plot vectors by using a set of rules established by the user, are useful for applications in the presentation of scientific information and in the development of graphical art. One useful application of user-supplied transformations is the contouring of data from irregular finite-difference grids. Transformation of plot vectors results in contours that retain the smoothness and detail of the original surface obtained from the finite-difference grid. An application in the development of graphical art is the transformation of straight lines into wavy or curved lines.

CHAPTER E--DATA BASES AND AUTOMATED DATA HANDLING

A VERTICAL SEQUENCE CORRELATOR MODEL (VERSECORR) WHICH IS USED IN THE RECOGNITION OF GEOPHYSICAL LOG SHAPES

by Merribeth Bruntz and A. Curtis Huffman, Jr.

ABSTRACT

Geophysical well log interpretation and correlation commonly are time consuming and subjective processes. Many studies involve analysis of hundreds of geophysical logs of differing types (such as, self-potential, resistivity, and gamma rays) and qualities. A pattern associator model has been developed that identifies a test set of specific log shapes. Neural computing techniques are used because of the large degree of generalization that must be performed in this identification.

Use of a neural network architecture that allows for the learning of 12 log shapes presented as part of the training set is demonstrated. Testing was performed on 20 patterns that are not part of the training set, with 15 of the test patterns succesfully identified.

INTRODUCTION

One of the most common forms of data collection and presentation is as x-y plots where intensity of a variable is displayed sequentially against time, distance, and depth. In the geological sciences, this type of plot has many applications, but perhaps the most familiar and frequently used is the geophysical well log. Since 1927, when the technique was invented in France, the well log has become an important analytical tool of the petroleum industry and is the basis of nearly any hydrocarbon exploration or development program. Although logging and analysis techniques have improved greatly, especially in the last 10 to 15 years, geologic interpretation and correlation of logs remain an art to a large extent and largely depend on the ability of the interpreter to recognize patterns.

With this in mind, the goal of this project is to develop a computer model that offers an objective interpretation of these patterns. Neural computing techniques are being used to explore the feasibility of using methods that simulate human intelligence more closely than those used in symbolic artificial intelligence. The acronym, VERSECORR, for Vertical Sequence Correlator, represents the requirement that the model recogize and interpret vertical sequences, and also correlate them with each other. In this paper we will discuss the design and implementation of a program that recognizes well log shapes, which is the first step in recognizing and interpreting vertical sequences.

NEURAL NETWORK ARCHITECTURE

Neural network models that perform specialized tasks, such as pattern recognition, have been successfully simulated on digital computers. Like the human brain, our model is composed of processing cells (neurons) and connections (synapses) between the processing cells. The processing cells function as memory units that contain pieces of information. The connections between the processing cells connect the pieces of information together, forming a network of neuron-like units , u_i and v_j (fig. E1).

Each connection in the network has a numerical weight, w_{ij}, that corresponds to the influence of unit v_j on unit u_i. Positive weights indicate reinforcement, and negative weights represent inhibition. A subset of units in the network, $v_1,....,v_n$, is considered as network inputs. These network inputs have no entering connections. The outputs of the network, $u_1,....,u_m$, form another subset of units in the network called output units.

The neural network distinguishes associations between input patterns and output patterns. In general, this type of model is called a pattern associator and is one of the simplest of neural models.

Each of the neuron-like units in the network, whether input or output, has a numerical value called an activation value. The activation value for an output unit, u_i, is based upon the activation values of the input units, v_j, directly connected to it and the corresponding weights for these connections. The activation values of the input units are discrete, taking on the values $\{+1, \text{ or } -1\}$. Every output unit, u_i, computes its new activation, $\hat{u}_i$, as a function of the weighted sum of the input to unit, u_i, from directly connected units. Thus,

$$S_i = \sum_{j=1}^{n} w_{ij} v_j$$

and,

$$\hat{u}_i = f(S_i).$$

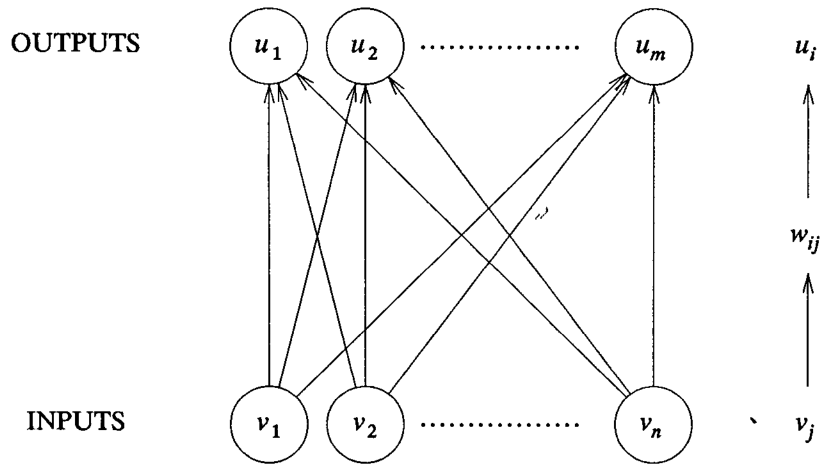


Figure E1.--Neural network architecture of a pattern associator model.

The activation of the input units are fixed by externally supplied input patterns. The activation value of a specific input unit, v_j, is determined by an interface that uses a linear least-squares approximation to determine features of the top, side, and bottom of a geophysical log shape. Some of these features include facing left or right, horizontal position, vertical position, or degree of slope (fig E2). The uniqueness of these features for a specific log shape is then used as the externally fixed input activations.

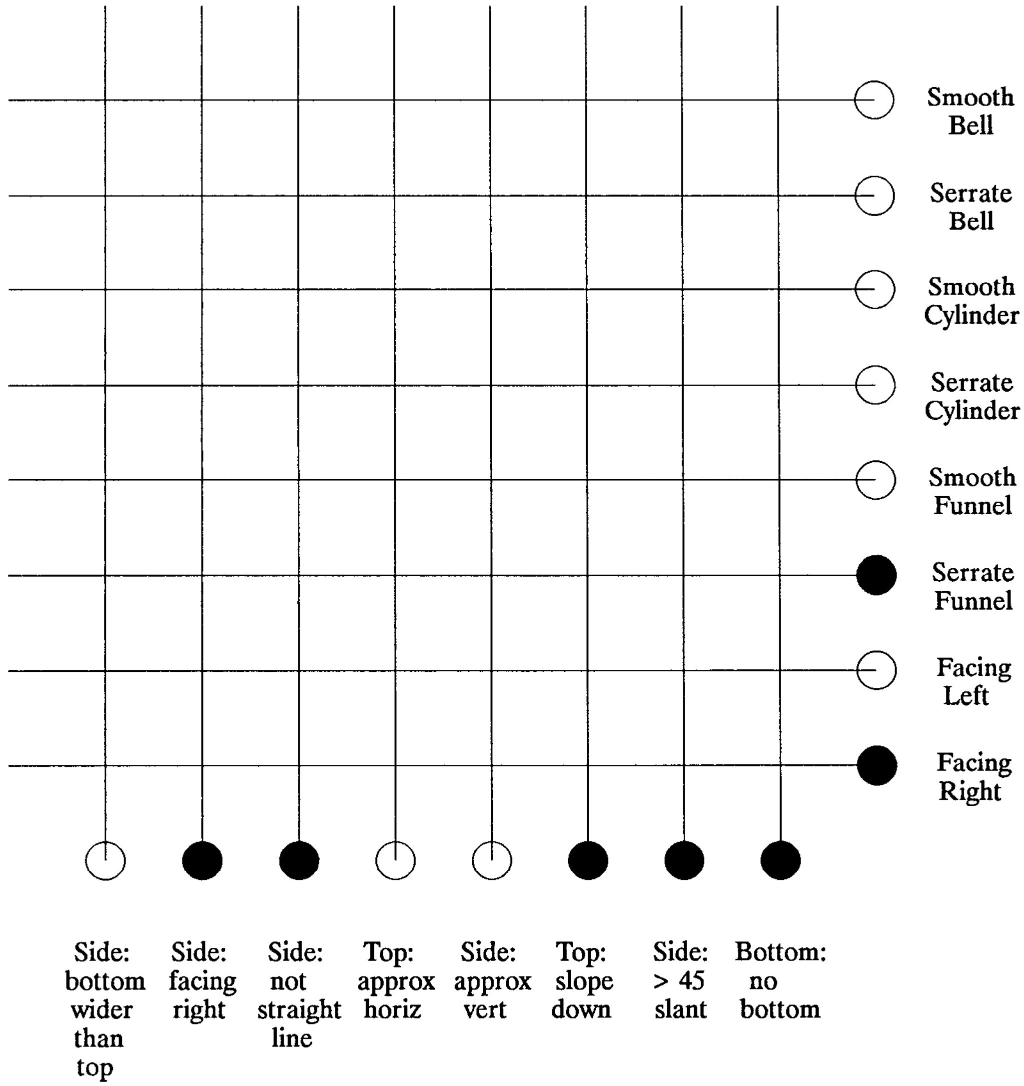


Figure E2.--Pattern associator model. The horizontal input units represent the features of the log shapes, and the vertical output units represent the shape identification. The solid units refer to a positive activation value for that unit, whereas the open units refer to a negative activation value.

LEARNING

Most neural network models use learning algorithms. In general, learning deals with finding the connection weights that will produce a desired response for the model. Usually, the learning algorithms work from training examples of desired response to generate appropriate weights, w_{ij}. Each training example consists of the desired, or target, activations for a set of output units when the input units are fixed to specific activation values.

The process of training the network by repeatedly presenting the same input patterns and their corresponding target output patterns, causes the system to self-organize by generalizing and adapting until the correct responses are produced. In our development of VERSECORR, we tested two methods of learning: the Hebb rule, and the delta rule.

The Hebb Rule

Donald Hebb, in 1949, published the definition of a learning process that explained the modification of synapses between neurons. His definition of what is known as the "Hebb synapse" states that when two units are activated at the same time the strength of the connection between them should be increased.

The mathematical formulation of the Hebb Rule can be stated as follows (Rumelhart and McClelland, 1986, p. 53):

$$\Delta w_{ij} = \eta a_i o_j \ . \tag{1}$$

As stated earlier, the network is trained by repeatedly presenting input-output pattern pairs. So, for any learning event, k,

$$w_{ij} = \eta a_{jk} o_{ik} \ . \tag{2}$$

In this equation η refers to the value of the learning rate parameter, a_{jk} refers to the activation of input unit j with input pattern $\vec{a}_k$, and o_{ik} refers to the activation of output unit i in output pattern $\vec{o}_k$. Each weight is the product of the activation of the input unit and the activation of the output unit in learning event k.

Kohonen (Kohonen, 1988) provides an analysis of the activation of an output unit in a specific learning event; this keeps with the notation established above,

$$o_{ik} = \sum_{j=1}^{n} w_{ij} a_{jk} \ . \tag{3}$$

If we present a test input pattern, $\vec{a}_t$, to the network, then the resulting activation of output unit i when tested with input pattern $\vec{a}_t$ is

$$o_{it} = \sum_{j=1}^{n} w_{ij} a_{jt} \ . \tag{4}$$

Substituting the value of w_{ij} from equation (2) yields

$$o_{it} = \sum_{j=1}^{n} \eta a_{jk} o_{ik} a_{jk} \ . \tag{5}$$

Finally, because we are summing with respect to j in equation (5), we have

$$o_{it} = \eta o_{ik} \sum_{j=1}^{n} a_{jk} a_{jt} \; . \tag{6}$$

The sum of products of corresponding elements is the dot product; this represents the similarity between the two patterns $\vec{a}_k$ and $\vec{a}_t$.

Because the patterns consist of only +1 or -1 values, the dot product corresponds to the correlation between the two patterns, $\vec{a}_k$ and $\vec{a}_t$, if the dot product is normalized (that is, divided by its length to produce a unit vector). This method is standard in the application of Hebbian learning.

The normalized dot product has a value of 1 if the patterns are identical, a value of -1 if they are exactly opposite of each other, and a value of 0 if the elements of one pattern vector are completely independent of the elements of the other. When the elements of one pattern vector are completely independent of the elements of the other, the patterns are said to be orthogonal. And, when all the members of a set of patterns are orthogonal to each other, they form an orthogonal set.

In Hebbian learning, if the input patterns used in training the network do not form an orthogonal set, then it is not possible to avoid "cross-talk" between the response that is appropriate to one output pattern and the response that occurs to the others (Rumelhart and McClelland, 1986, p. 63).

The Delta Rule

The delta rule, also known as the Widrow-Hoff learning rule, allows us to overcome the orthogonality limitations of the Hebb rule. This is done by using the difference between the target activation and the obtained activation to drive learning. The goal is to adjust the strengths of the connections so that they will tend to reduce this difference measure.

The delta rule for a particular input-target pair $\vec{a}_p, \vec{t}_p$ can be stated as (Rumelhart and McClelland, 1986, p. 322):

$$\Delta w_{ij} = \eta(t_{ip} - o_{ip})a_{jp} \; . \tag{7}$$

Thus, the weights that result from a group of learning pairs indexed by p can be written as:

$$w_{ij} = \eta \sum_{p} (t_{ip} - o_{ip})a_{jp} \; . \tag{8}$$

If the input patterns used in training the network form an orthogonal set, then the delta rule produces the same results as the Hebb rule. If, however, the input patterns used in training the network do not form an orthogonal set, then (by equation 7) the change in the weights that occurs on a learning trial stores an association of the input pattern with the error pattern. That is, an increment is added to each weight that can be thought of as an association between the error for the output unit and the activation of the input unit.

For example, if we have an input pattern, $\vec{a}_p$, that is paired with a target pattern, $\vec{t}_p$, on the output produced by test pattern $\vec{a}_t$, then, by equation (7), the effect of the change in the weights due to a training session will change the output of some output unit i by an amount proportional to the error that occurs for that unit during the training session. This error, ε_i, is multipled by the dot product of the learned pattern with the first test pattern. This is illustrated in the following equation:

$$\Delta o_{it} = n\,\eta\varepsilon(a_p \cdot a_t)_n \tag{9}$$

where n equals the number of input units.

Thus, the change in the output pattern at test is proportional to the error vector times the normalized dot product of the input pattern that occurred during learning and the input pattern that occurred during test.

SYSTEM TRAINING and TESTING

By use of the delta rule to drive learning, it is possible to train VERSECORR to recognize the geophysical log patterns. The patterns (fig. E3) used in the training of VERSECORR do not form an orthogonal set; this makes it impossible to use Hebbian learning in the training of the network. Therefore, the network is trained using the delta rule to drive learning.

The input-output pattern pairs representing the log patterns (fig. E3) can be found in table E1. The input patterns are based on the presence or absence of a specific input feature, derived by the interface. The output patterns follow from the combination of the input features leading to the a specific output identification.

Table E1.--Training set of input-output pattern pairs

Pattern name	Input	Output
Smooth bell (left)	+ - - - - + - -	+ - - - - - + -
Smooth bell (right)	+ + - - - + - -	+ - - - - - - +
Serrate bell (left)	+ - + - - + - -	- + - - - - + -
Serrate bell (right)	+ + + - - + - -	- + - - - - - +
Smooth cylinder (left)	- - - + + - - -	- - + - - - - +
Smooth cylinder (right)	- + - + + - - -	- - + - - - - +
Serrate cylinder (left)	- - + + + - - -	- - - + - - + -
Serrate cylinder (right)	- + + + + - - -	- - - + - - - +
Smooth funnel (left)	- - - - - + + +	- - - - + - + -
Smooth funnel (right)	- + - - - + + +	- - - - + - - +
Serrate funnel (left)	- - + - - + + +	- - - - - + + -
Serrate funnel (right)	- + + - - + + +	- - - - - + - +

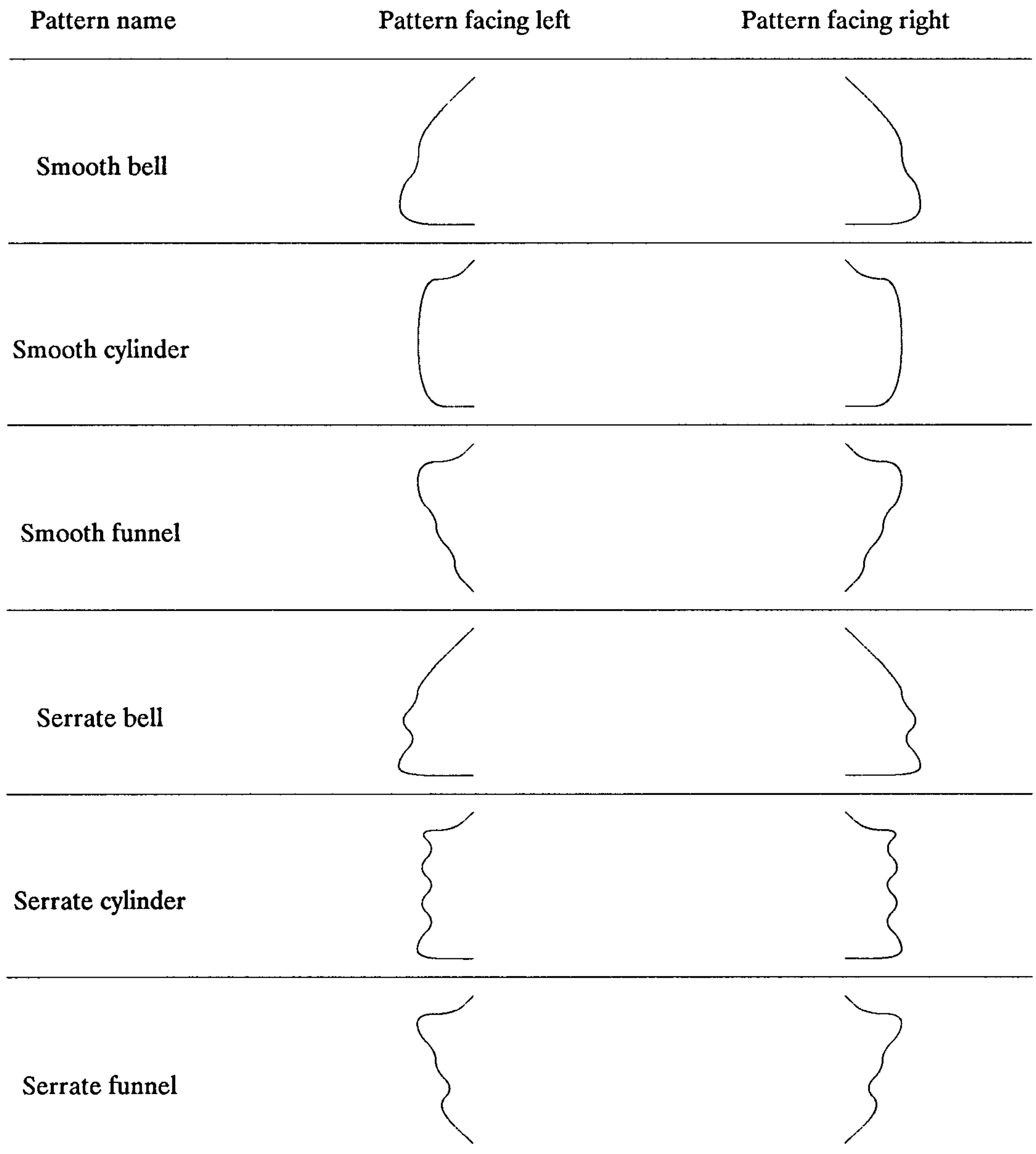


Figure E3.--Training set of log shapes

Upon completion of the network training using the training set shown in table E1, a test set of patterns were input. Figure E4 shows the test set of input patterns and the resulting output identifications.

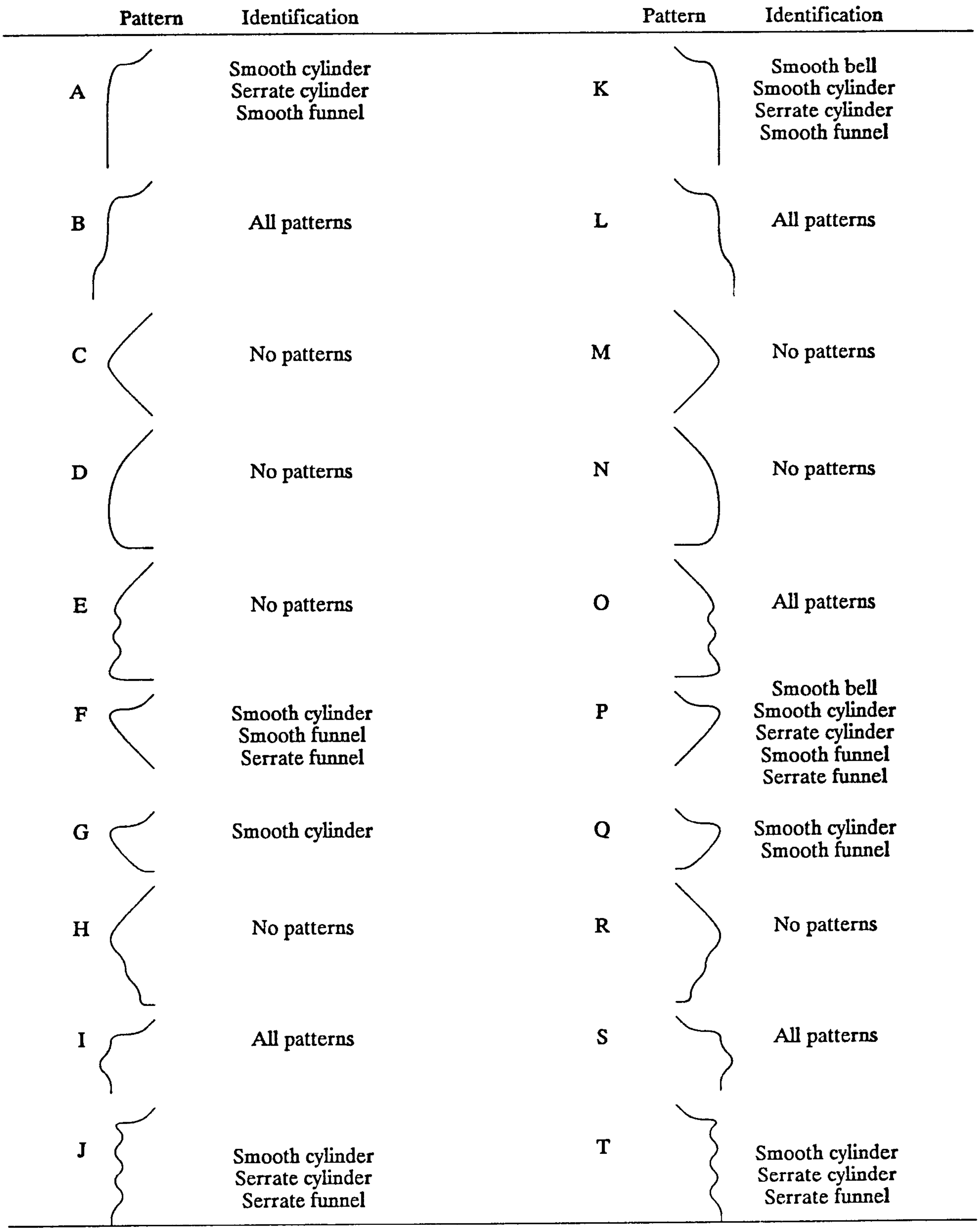


Figure E4.--Test set of log shapes

RESULTS of TESTING

VERSECORR performed reasonably well in identifying correctly 15 of 20 patterns that were not part of the training set. There were, however, 5 cases (D, E, H, K, and N of fig. 4) in which VERSECORR made incorrect identifications. The system's inability to produce consistent output in these cases may have been caused by the use of a linear least-squares interface to produce the input patterns. It is possible that in some cases this method produces an input-output pattern similarity structure that is too different.

Our problems might also be solved by adding intermediate layers of hidden units that create their own internal representations of the input units. Hidden units are units whose inputs and outputs are within the neural network model. These units are not influenced by anything external to the model, such as an interface program. Moreover, for VERSECORR to recognize sequences of patterns, the addition of intermediate layers of hidden units will likely be required. This will also lead to the use of more robust learning algorithms.

CONCLUSIONS

VERSECORR is a pattern associator network that maps a set of input layer patterns directly to a set of output layer patterns. In many cases, patterns associator networks make reasonable generalizations and perform resonably on patterns that have never before been presented to the system. This is because pattern associator networks map similar input patterns to similar output patterns. The similarity of patterns is determined by their overlap. The overlap in pattern associator networks is determined by whatever produces the patterns outside the learning system.

The elementary neural computing techniques used by VERSECORR show considerable promise in pattern interpretation. As a result, continued development of these techniques is needed in future research into vertical sequence interpretation and correlation problems.

REFERENCES CITED

Kohonen, T., 1988, *Self-organization and associative memory:* New York, Springer-Verlag, 312 p.

Rumelhart, D. E., and McClelland, J. L., 1986, *Parallel distributed processing (v. 1 and 2):* Cambridge, MA, MIT Press, 1158 p.

COMPUTER PROGRAMS FOR PROCESSING MODEL DATA AND RESULTS FOR STEADY-STATE AND TRANSIENT GROUND-WATER MODELS

By Carmen R. Baxter

U.S. Geological Survey

ABSTRACT

Several computer programs have been modified or developed by the U.S. Geological Survey to analyze the results of steady-state and transient-state data ground-water model simulations and to modify the input data for the models to develop the best possible fit between computed and observed conditions. Computer programs designed for modification of input data, such as pumpage and riverbed conductance is useful in evaluating the sensitivity of simulated aquifer properties and related parameters. Programs that redistribute data over given locations in a study area help to better address site-specific problems. Tables and hydrographs also are effective in examining time-variant output data at selected grid locations produced by transient simulation. Model output also can be used as input data for graphics computer programs for producing linear or spatial plots that may include grids enhanced with posting, contouring, or other data indicators.

INTRODUCTION

Performing model calibrations and sensitivity analyses on large amounts of data can prove to be cumbersome using text editors. A lack of consistency is a common occurrence in data modified directly by the modeler. Pre- and post-processing programs are required to analyze the results of steady-state and transient runs or to modify the input data to develop the best possible fit between observed and computed conditions. Several software techniques to interpret the results of model runs or to modify input data are currently employed in the Arkansas District of the U.S. Geological Survey. These techniques have been used to process data and to evaluate the hydrology in the East Arkansas Comprehensive Study (EARCS) project area.

Purpose and Scope

The purpose of this paper is to give examples of computer programs that have been used in the Arkansas District for the processing of model data that may be beneficial to other district personnel in need of pre- and post-processing methods on model data. The examples include overviews on global modification and data redistribution programs for reprocessing of model input. For post-processing of model output, overviews of programs using nongraphic- and graphic-oriented programs with their output are presented. Examples include tables and hydrographs produced for the purpose of calibrating the model program's input parameters as are sensitivity analyses and grid overlays of maps using the graphics packages DISSPLA and TELLAGRAF.

Background Information

The pre- and post-processing computer programs discussed in this paper were developed using Fortran 77 and the graphics packages DISSPLA and TELLAGRAF (Integrated Software Systems Corp., 1984a; 1984b). The modular three-dimensional finite-difference ground-water flow model program (McDonald and Harbaugh, 1988) was used to produce steady-state and transient-state model simulations to a ground-water flow system of a study area.

Model calibrations define aquifer parameters, boundary conditions, and other parameters in order with the aid of a model program, for known conditions to be simulated in the modeled system. The modeler may have to run the model program repeatedly, making reasonable changes to given parameters that he feels are less reliable to produce results that adequately match observed field data.

Sensitivity analyses which are performed after the final model calibration help the modeler to evaluate the reliability of the model. To make the analyses, the modeler first changes the values of selected input parameters; usually, a certain percentage change of the calibrated value is made for one parameter at a time.

The programs developed to perform model calibrations and sensitivity analyses were implemented on a ground-water flow system model of the East Arkansas Regional Comprehensive Study (EARCS) project area. The purpose of the EARCS was to digitally simulate the current conditions in the Mississippi Valley alluvial aquifer and to project the future consequences of increased water demands placed upon the aquifer. The study area is located in east-central Arkansas and encompasses a 13,400 mi^2 (square mile) area, including all or parts of 24 counties. Two grid-cell systems based on the Universal Transverse Mercator System (UTM) were developed. One grid has 86 rows and 52 columns with each grid node called a 3-mile cell that represents a 9 square mile area. The second grid overlaying the same area as the first has 258 rows and 156 columns with each grid node called a 1-mile cell that represents a 1 square mile area.

REPROCESSING OF MODEL INPUT DATA

Input parameters for the model program are processed continually to optimize the computed heads. Once a final model calibration is made, sensitivity analyses are performed to evaluate the reliability of the model and to check the sensitivity of selected location.

Global Modifications

The most common method of modification to input parameters is to increase or decrease all values within a given file by a specified percentage. The change in model-computed heads following such a change in an input parameter will give insight to the sensitivity of the model to that parameter at various locations in the study area. A program that was developed as a part of EARCS allows the user to enter a percentage to be applied to a given input file from which a new file is created for model test runs. Another modification to the program permits different percentages of change in input parameters to be applied to each stress period.

The above program that applies an overall percentage to an input file was modified with a subroutine using the boundary array, a model input file, which allowed the user to enter a percentage for each river basin. The boundary array is a node-level map that allows the analyst to define the status of each grid node and to manipulate parameter input to the selected locations. This array contains integer codes defining the regions in the study area. Zeroes indicate inactive cells, negative values are constant heads, and positive values indicate river basins, rivers, and other geographic features important to the processing of the model. The boundary array is useful in indicating problem regions in a study area.

Data Redistribution

Data is redistributed by taking a value from a given cell location from one grid overlay and distributing the value into cell locations of another grid overlay covering the same spatial area. Two grids were used to overlay the same study area for the EARCS project: (1) a grid consisting of 86 rows and 52 columns with each cell being 3 miles on a side (9 mi^2), and (2) a grid consisting of 258 rows and 156 columns with each cell being 1 mile on a side (1 mi^2). The 3-mile grid gave an approximate regional look at the study area and provided the initial optimization data early in the project. Subsequently, a 1-mile grid was used to better address site-specific problems in the area. A data file containing river recharge values for active nodes in the 3-mile grid had been developed and a new file of river recharge for the 1-mile grid was required. A program was written that used the boundary array and the river-recharge file of the 3-mile grid and the boundary array of the 1-mile grid to produce the new values. The output river recharge to a 1-mile cell was based on the following conditions:

1. For a 3-mile basin node, the river recharge value is placed in all active 1-mile cells within 3-mile node.
2. For a 3-mile river node, its river recharge value is placed into every 1-mile river node equivalent to the 3-mile boundary code. The remaining 1-mile active cells retrieve their values from the first adjacent active cell checked that has a nonzero value. The adjacent cell may be part of the 3-mile node or an adjoining 3-mile node.
3. For a nonactive 3-mile node, any active 1-mile cell within the 3-mile node will retrieve its value from the first adjacent active cell checked with a nonzero value and the nonactive cells are set to zero. The adjacent cell may be part of the 3-mile node or an adjoining 3-mile node.

The resulting array for the 1-mile recharge was then written to a file to be used as input for the 1-mile model.

POST-PROCESSING OF MODEL OUTPUT

Model programs give the user a limited control over the amount of output and sometimes the form of the output that is generated. Depending on whether the modeler is calibrating the model or performing sensitivity analyses, he may wish to view the output in other forms that will best help him analyze the data. Some output forms would include tables, hydrographs, and grid.

Tables are commonly used in processing of model data to enable the user to view information from selected grid locations in an uncomplicated form. One or more sets of data may be displayed in a table for listing or comparative purposes. Most tables have a simple format consisting of a heading that describes the information to be listed and subheadings that indicate the value(s) and the respective row and column location.

Hydrographs and grids also are used in post-processing to analyze the results of a model run. Hydrographs may be produced with or without graphics programs to illustrate sensitivity analyses. Contouring, posting, and other forms of data indicators often are placed on graphically produced grids to give the modeler a visual aid for locating problem areas or indicating geographical features in the study area.

Tables and Hydrographs for Examining Data at Selected Grid Locations

Tables have been used to compare computed and observed water levels at selected locations in the study area. A tabling program was developed to compare the contents of a file containing selected observed water levels of a given stress period with the computed heads of the same stress period. A difference between the computed and observed heads then is calculated. The selected differences, computed heads, observed heads, and their corresponding locations are written on a single table. Further modification to the program allows a table to be produced from selected nodes containing the computed and observed water levels and their difference for all stress periods.

This tabling program was then expanded to include hydrographs. The purpose of the hydrographs is to provide a pictorial representation of the tabled data and to allow the modeler to examine the fit between the two unconnected sets of points marking the integer form of the observed and computed water levels. The symbols used to mark the water levels at a given stress period are as follows:

C -- computed water levels,
O -- observed water levels, and
* -- overlapping of computed and observed water levels.

Note that the "overlapping" of computed and observed water levels is the integer form of the values, not the decimal values. For example, if the observed value for a given stress period is 185.00 and its corresponding computed value is 185.75, then both values would be plotted to the hydrograph as 185.

The coordinates of the graph are based on the stress period and the water level in feet above sea level. The range of the water-level axis is further reduced by determining the minimum and the maximum of the levels. The minimum is lowered 10 feet and the maximum is set 110 feet above the new minimum before the two values are used as the beginning and end labels of the axis. The program also constructs the table and hydrograph for a selected location to be printed on one page if the user spools to a line printer using the spool option -FTN (fig. E3).

Input Data to Graphics Software

Model output is used regularly to produce linear plots for sensitivity analyses. The most common linear plot is the hydrograph which portrays the data results in head files produced from steady-state and transient model runs. The data compared to the results may be observed data collected for a given time period or other model output that may vary from the final calibrated results as obtained by modifying pumpage, riverbed conductance, or another input parameter. Graphics packages available such as DISSPLA and TELLAGRAF are useful in illustrating these comparisons (Integrated Software Systems Corp., 1984).

Hydrographs

The tabling program that produced line-printer hydrographs for calibration purposes was modified further to produce input data for TELLAGRAF. The user enters the run number of the head file to be processed and a separate file is created for each selected location. Each file contains a list of observed water levels and a list of computed water levels for all stress periods for a given location. After execution of the program, these files are loaded separately into a TELLAGRAF data file and saved under different file names. Each new TELLAGRAF data file is entered into the TELLAGRAF program. Enhancements to the data file were made from within TELLAGRAF to produce a hydrograph at a given row and column location. This hydrograph shows calibrated water levels as compared to observed water levels for seven stress periods (fig. E4).

Other programs were written using DISSPLA routines for sensitivity analyses. One program allows the user to indicate whether a sensitivity analysis is to be performed on files produced by a steady-state or transient model run. If a transient condition is chosen, the user enters the row and column to be analyzed. The hydrograph produced has an axis of water levels and an axis of stress periods. If a steady-state condition is selected, the user is prompted for a row and range of columns to be analyzed. The hydrograph created has an axis of water levels and an axis consisting of the range of columns entered by the user. The user then enters the final calibrated head file and up to four additional test head files that can be compared to the contents of the final head. The user also can indicate whether a title is to be printed; an explanation is to be printed; and the curves drawn are to be of different colors or all black.

Two other programs performed sensitivity analyses on steady-state models only. The programs are identical except one performs an analysis on a given row and the other program analyzes a given column. The name of the final

STP	LOC	OBSERVED	COMPUTED	DIFFERENCE
1	1537	187.00	186.52	0.48
2	1537	185.00	185.96	-0.96
3	1537	181.00	183.71	-2.71
4	1537	177.00	181.47	-4.47
5	1537	171.00	173.88	-2.88
6	1537	164.00	162.56	1.44
7	1537	161.00	150.46	10.54

EXPLANATION

O = OBSERVED VALUE
C = COMPUTED VALUE
* = OVERLAPPING OF OBSERVED AND COMPUTED VALUES

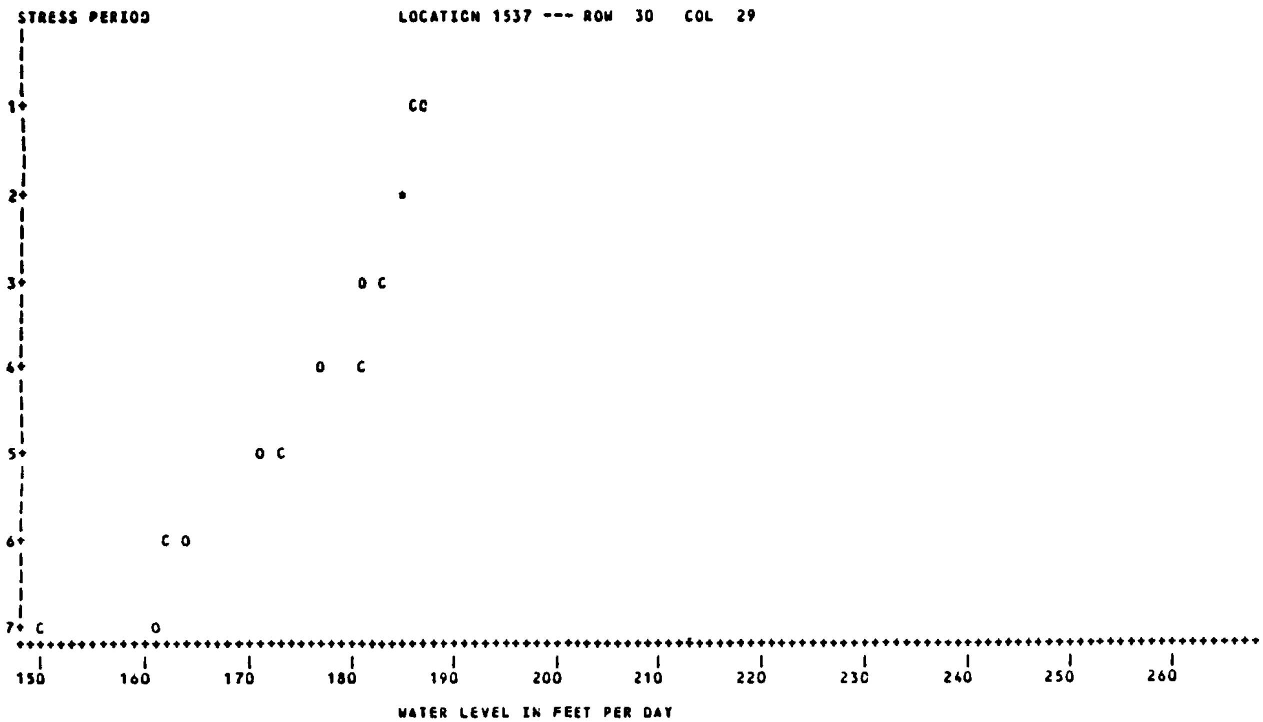


Figure E3. -- Table and hydrograph for transient calibration simulations.

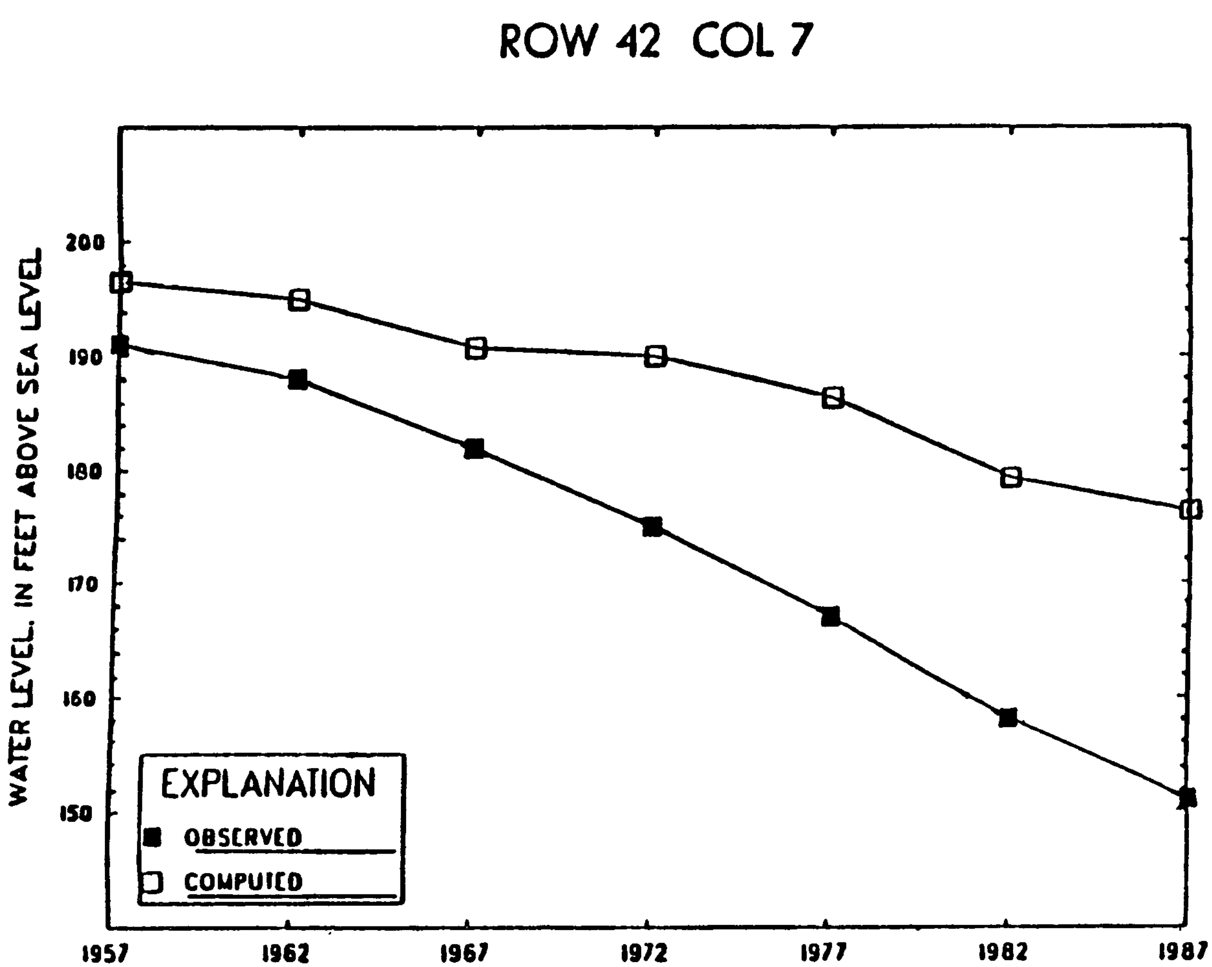


Figure E4. -- Hydrograph showing computed water levels produced from a transient model simulation as compared to observed water levels for seven stress periods.

calibrated head was coded in the programs. The user has to enter only four test files with descriptions for an explanation. The user then enters the row (or column) to analyze and the device by which the plot is to be drawn. The values marked on a plot (with a difference limit of plus or minus 40 feet) are the nodal value differences between the calibrated head and the test file. If the difference is greater than 40 feet, then the difference plotted is 40 feet. If the difference is less than -40 feet, then the difference is -40 feet. Figure E5 shows an example of such a sensitivity analysis for a given row.

Grids

A grid is a group of intersecting lines that form squares and is used as a reference for locating points in a study area. Data can be placed in the grid cells to give the modeler a visual aid for locating problem areas or indicating geographical features in the study area.

Posting

To understand the distribution of model data, one may post or print numerical values to a grid that will overlay a map of the study area. Posting data values will allow the checking of input data errors by the modeler, show the distribution of the output over various physical features in the study area, and indicate the problem regions.

A program incorporating DISSPLA routines to produce a grid overlay with 86 rows and 52 columns was developed specifically for the EARCS area. The grid was created at a 1:500,000 scale to be placed over a Lambert Conformal Conic map projection. Four latitude and longitude reference marks were drawn on the grid and a region of nonactive nodes at the bottom right was blanked for the grid title, latitude and longitude of the origin, and other information. Additional data can be added manually by the modeler or placed on the grid by a cartographer. Such data may include county codes and boundaries or the codes from the boundary array.

To permit the modeler to examine model input and output on the grid, the program was modified to query the user for a descriptive message about the data and an input file containing an array of values to be posted. An example of a posted grid is shown in figure E6. The values posted are saturated thicknesses produced from a transient run. A projected pumpage file divided into six stress periods (1990 to 2040) was used as pumpage input to the ground-water flow model and the resulting heads were manipulated to produce the saturated thickness for each cell in a given stress period.

Another available program produces grids at various scales. This program is designed specifically to be modified internally by the user to meet grid requirements. Upon execution of the run file, the user enters his boundary array file and indicates whether or not to post a data file. Other prompts include terminal type, paper size, and plot title. Therefore, the user can produce a base grid with only boundary array indicators or a grid with posted values.

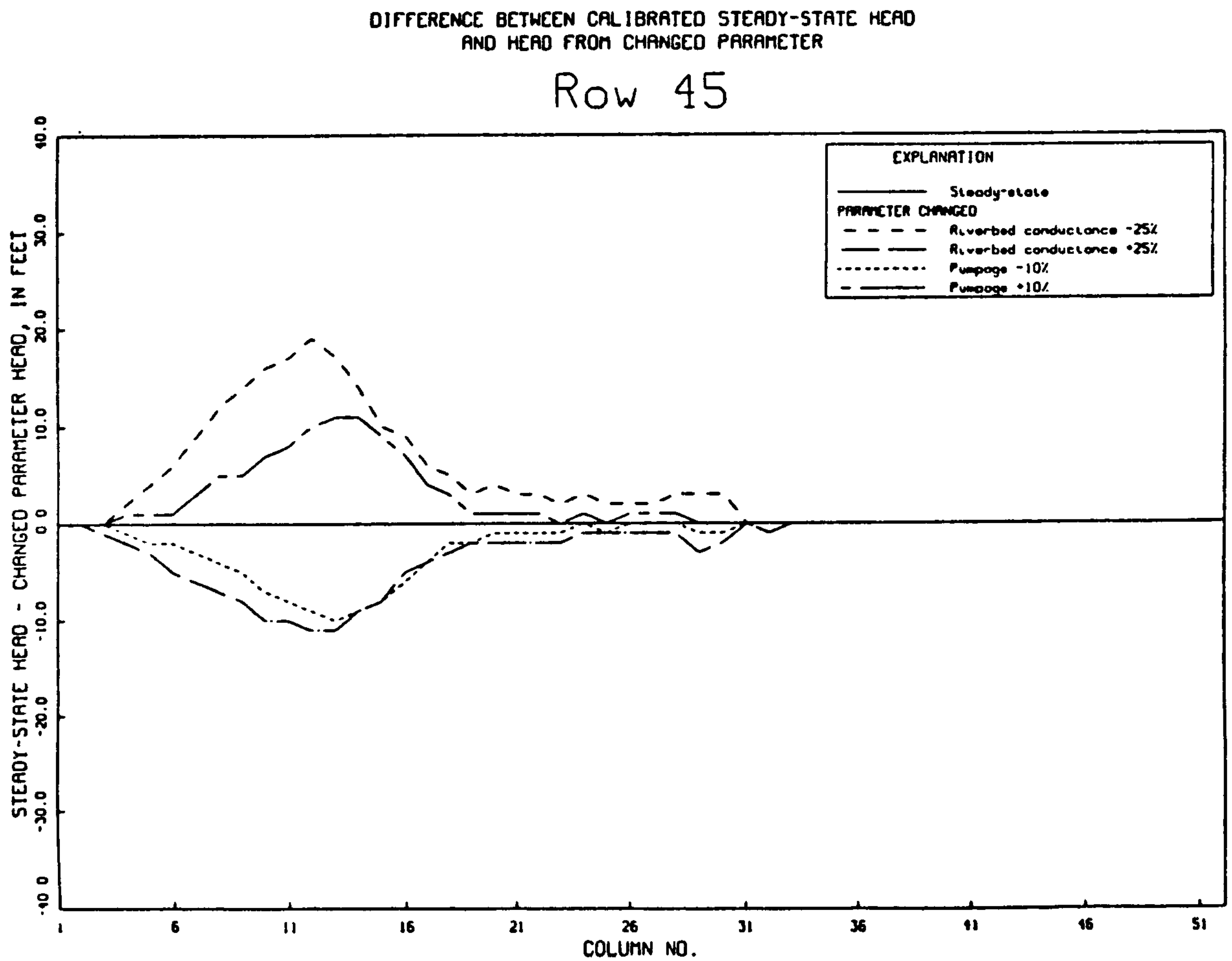


Figure E5. -- Hydrograph showing a sensitivity analysis of the difference between calibrated steady-state head and heads produced by changing other parameters.

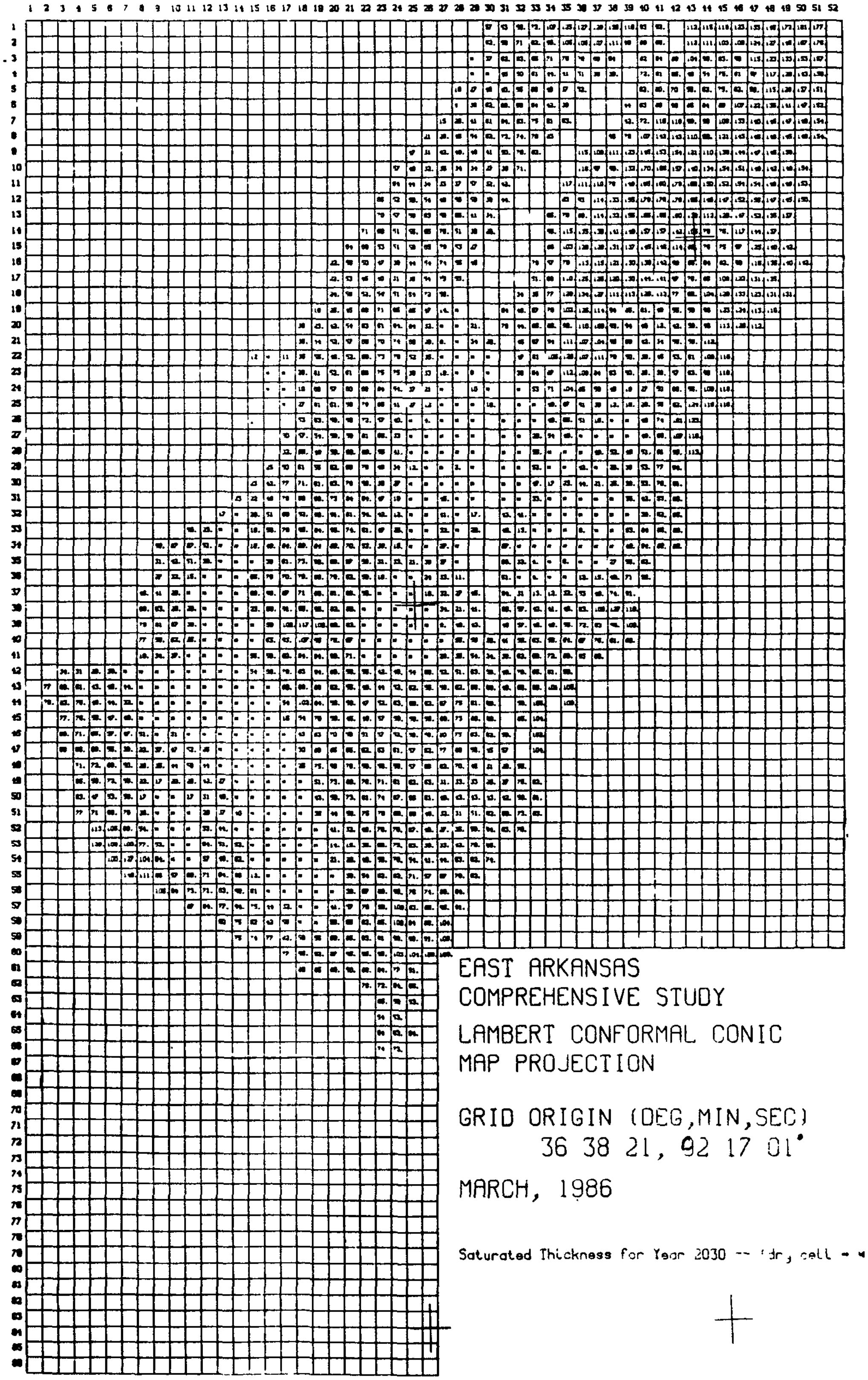


Figure E6. -- Posted values of saturated thickness projected for the year 2030.

Contouring

Contouring can be an additional feature to grids. Contour lines connect points of equal elevation and show the configuration of the surface. The posting program which produced grids at various scales, was modified with more DISSPLA routines for contouring data sets. DISSPLA generates contour lines by linear interpolation of adjacent grid cells in both the X and Y directions. With this enhancement, the modeler now has the option to post and (or) contour.

A contouring example is the potentiometric surface, which after the plotting and contouring of water-level altitudes, is the hydraulic head of an aquifer. The contour lines on the map indicate the direction of the ground-water flow that is perpendicular to the contours through the aquifer. A grid with a potentiometric map of the 1982 water levels in eastern Arkansas is shown in figure E7.

Other Data Indicators

Representative characters can be used to emphasize specific data values, ranges, or other characteristics on a grid. Additional changes to the posting routine allows certain ranges of data to be marked on the grid. The projected saturated thickness of 20 feet or less for the year 2030 is indicated with an "X" on the grid, thus allowing the modeler to note critical regions in the study area (fig. E8). One may note that this routine already emphasizes various codes from the boundary array. Asterisks (*) were inactive nodes, R indicated river cells, and C represented nodes in a hydrologic barrier called Crowleys Ridge.

SUMMARY

Several software techniques are presented that modified input or analyzed results of the McDonald-Harbaugh ground-water flow model. Two examples of reprocessing model input are given: global modification and data redistribution. For the post-processing of model output, nongraphic and graphic techniques are given. Model output for calibration purposes is presented in tables and line plotter hydrographs. Graphic routines execute sensitivity analyses and produce grid overlays with posting, contouring, and other data indicators.

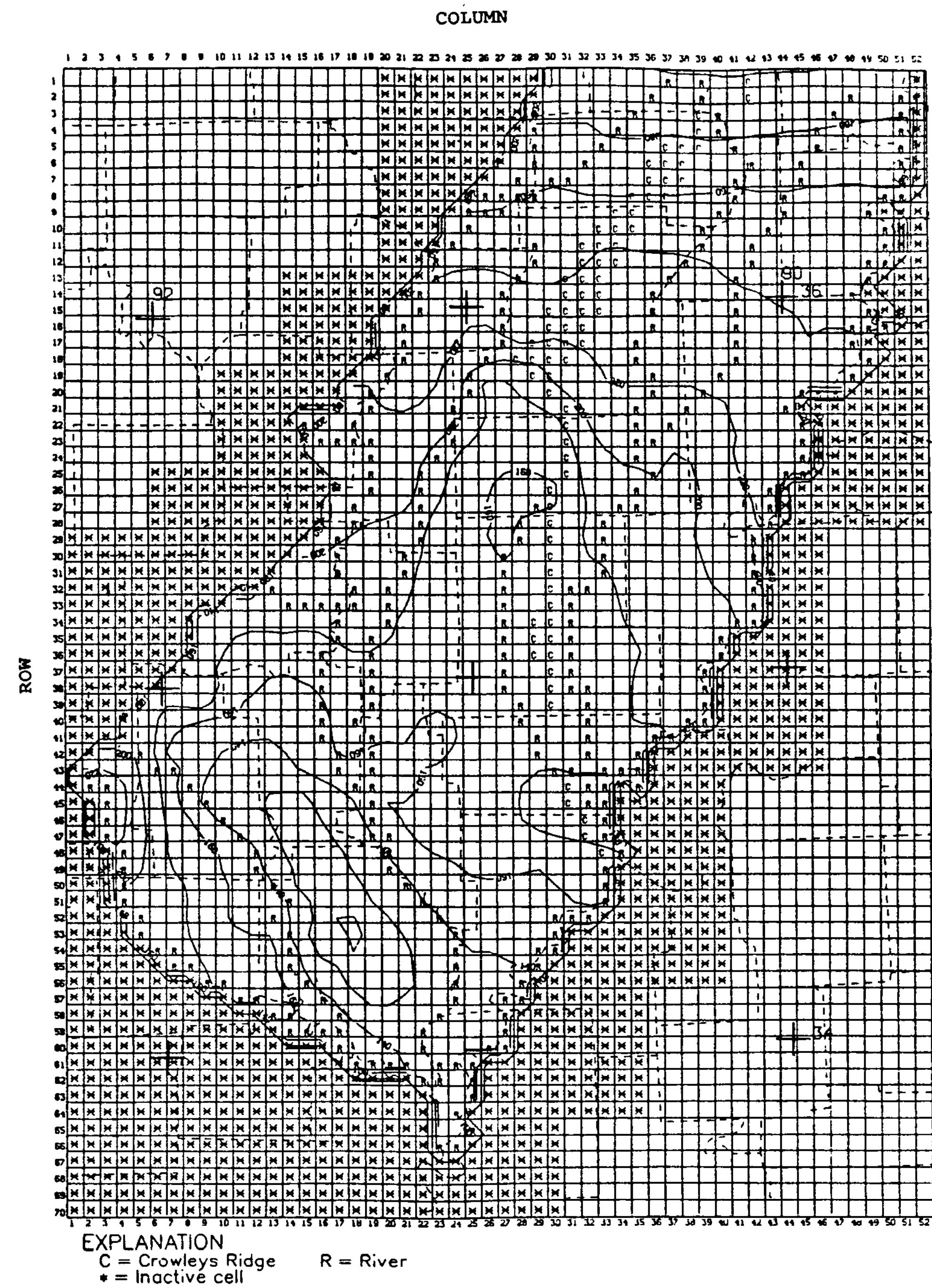


Figure E7. -- Potentiometric map of the 1982 water levels in eastern Arkansas.

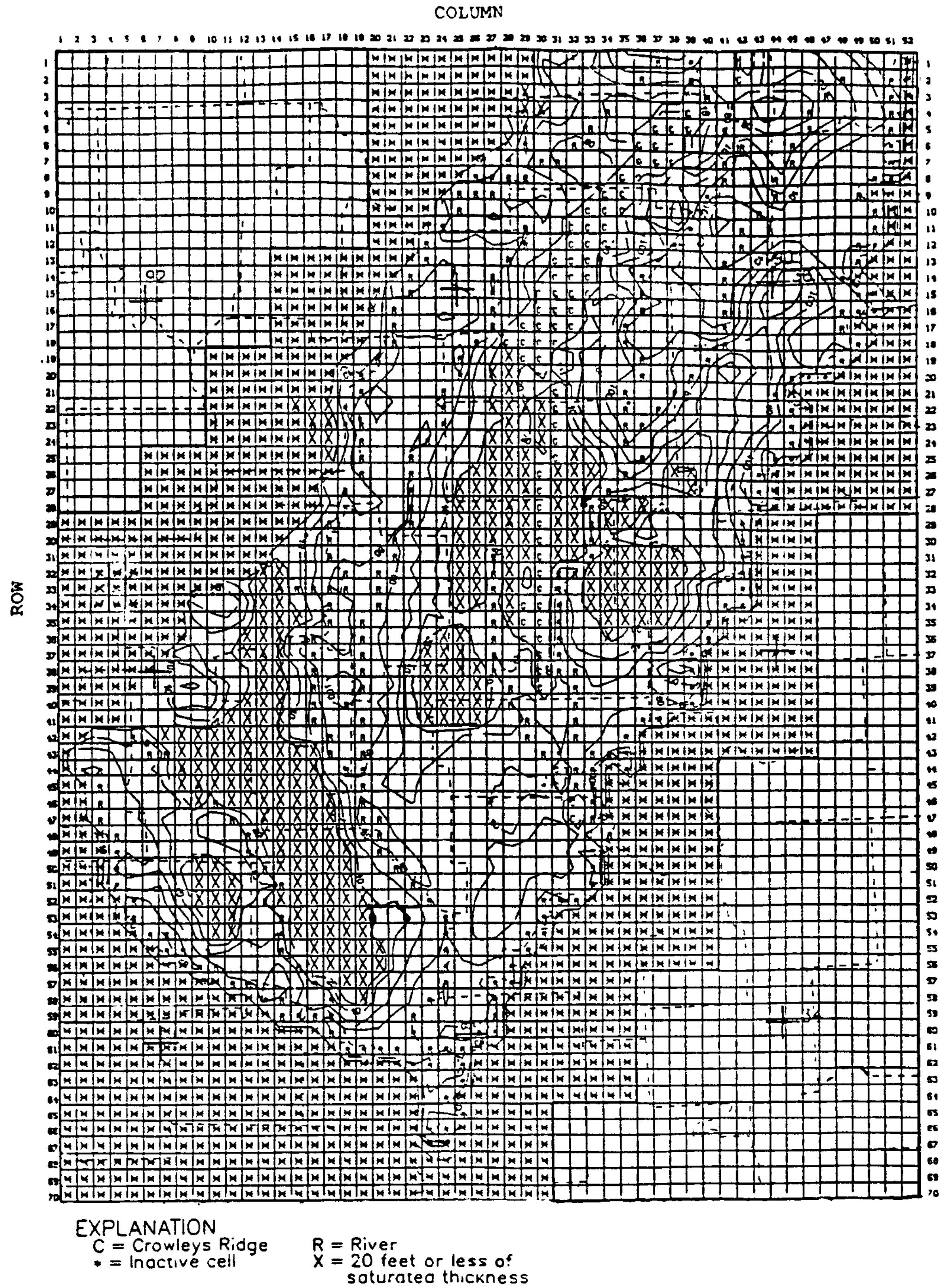


Figure E8. -- An example of data indicators by the marking of cells with 20 feet or less of saturated thickness.

SELECTED REFERENCES

Integrated Software Systems Corporation, 1984a, DISSPLA User's Manual, version 10.00.

----- 1984b, TELLAGRAF User's Manual, version 6.00.

Landy, Alice, 1982, CPL User's Guide, Revision 19.0, Prime Computer, Inc.

Lohman, S.W., Bennet, R.R., Brown, R.H., Cooper, H.H., Jr., Drescher, W.J., Ferris, J.G., Johnson, A.I., McGuiness, C.L., Piper, A.M., Rorabaugh, M.I., Stallman, R.W., and Theis, C.V., 1972, Definitions of selected ground-water terms -- revisions and conceptual refinements: U.S. Geological Survey Water-Supply Paper 1988, 21 p.

McDonald, M.G., and Harbaugh, A.W., 1988, A modular three-dimensional finite-difference ground-water flow model: U.S. Geological Survey Techniques of Water-Resources Investigations, Book 6, Chapter A1, 548 p.

A COMPUTER METHOD FOR ESTIMATING GROUND-WATER CONTRIBUTION TO STREAMFLOW USING HYDROGRAPH-SEPARATION TECHNIQUES

By Ronald A. Sloto
U.S. Geological Survey

ABSTRACT

A computer program used to estimate the ground-water component of streamflow (base flow) using hydrograph-separation techniques provides a rapid method of estimating base flow and eliminates subjectivity inherent in manual graphical methods. Three hydrograph-separation techniques are used: fixed interval, sliding interval, and local minimum. Input to the program is daily mean stream discharge in conventional daily values card-image format from U.S. Geological Survey data bases. The program provides printed statistical output, plot files, and base flow in card-image format. Printed statistical output includes monthly and annual summary information about the hydrograph separations, a summary of annual base flow and median base flow for period of record, frequency distribution of base flow, and percentage of ground-water contribution to streamflow. Plot files produce streamflow and base-flow hydrographs and frequency-distribution plots of annual base flows for period of record. Base flow estimated by computer techniques is comparable with base flow estimated by manual base-flow-recession and curve-fitting methods.

INTRODUCTION

Hydrograph separation divides streamflow into its two component parts: overland runoff and ground-water discharge (base flow). Hydrograph separation is an important technique used in a variety of water-resource studies. Hydrograph-separation techniques are commonly used to estimate ground-water contribution to streamflow and to define ground-water/surface-water relations. Hydrograph-separation techniques are also used to calculate hydrologic budgets and recharge rates.

The separation of a hydrograph into overland runoff and ground-water components is difficult and inexact, and the results are subject to different interpretations. Manual and computer techniques estimate base flow; no technique provides exact volumes of runoff and ground-water discharge.

Hydrograph separation generally is performed manually by graphical methods. Two commonly used manual methods are fitting a curve under a streamflow hydrograph (Linsley and others, 1982, p. 210) and base-flow-recession methods (Olmsted and Hely, 1962; Riggs, 1963; Rorabaugh, 1963). Quick estimates of base flow are sometimes obtained by sketching a curve below the streamflow hydrograph. Different hydrologists using the same manual hydrograph-separation method usually produce different estimates of base flow. The use of a computer program removes subjectivity inherent in manual methods. It substantially reduces the time required for hydrograph separation. For separation of a hydrograph for a long-term streamflow station with 90 years of record, manual methods take weeks, and computer methods take seconds.

The purpose of this paper is to briefly describe the hydrograph-separation program and the savings in time and elimination of subjectivity attained by its use. A comparison of base flow estimated by manual and computer techniques is presented.

DESCRIPTION OF COMPUTER PROGRAM

The hydrograph-separation program uses the three techniques of Pettyjohn and Henning (1979) to separate the ground-water and runoff components of a streamflow hydrograph: fixed interval, sliding interval, and local minimum. All three techniques are run simultaneously so that differences in base flows can be compared.

Input Requirements

Input to the program is in a conventional format from the U.S. Geological Survey National Water Data Storage and Retrieval System (WATSTORE) and Automated Data Processing System (ADAPS) data bases. Input is daily mean stream discharge in daily values card-image format. The program will accept data in water (October 1 to September 30) or calendar (January 1 to December 31) years. The program reads as little as 1 year of record or as much as the period of record for a station. The program accounts for gaps in the period of record; however, input must be complete water or calendar years of data.

Program Output

The program produces printed statistical output, plot files, and daily values for base flow in card-image format.

The printed output provides information on the quantity of base flow and the percentage of streamflow as base flow. This information is used in hydrologic studies to calculate water budgets and characterize ground-water/surface-water relations. The user selects short or long format printed output. The short format prints program warning messages and a summary of annual base flow and base-flow frequency by water and calendar year. The annual summary lists annual base flow and median base flow in inches, cubic feet per second, and million gallons per day per square mile and gives the percentage of streamflow as base flow. In addition to warning messages and an annual summary, the long format also prints monthly values of mean and total discharge, mean and total base flow, and percentage of streamflow as base flow for each year of record. Table E4 is part of the printed output that summarizes estimated annual base flow for French Creek near Phoenixville, Pennsylvania for the period of record.

Graphical output is used to compare the three separation techniques and to present base-flow frequency data. Graphical output consists of TELAGRAF plot files. Annual water year or calendar year streamflow and base-flow hydrographs can be plotted for the three hydrograph separation methods (fig. E14). The base-flow frequency distribution for the period of record for water or calendar years can be plotted for one to three separation techniques (fig. E15).

COMPARISON OF COMPUTER-SEPARATION TECHNIQUES

Hydrographs were separated into base-flow and runoff components for 1961-85 water years

Table E4.--Base flow for French Creek near Phoenixville, Pennsylvania, 1969-85
[In., inches; ft^3/s, cubic feet per second; $(Mgal/d)/mi^2$, million gallons per day per square mile; percent, percent of streamflow as base flow]

WATER YEAR BASE FLOW SUMMARY FOR FRENCH CREEK NEAR PHOENIXVILLE, PA

Water	Fixed interval method				Sliding interval method				Local minimum method			
year	In.	ft^3/s	$(Mgal/d)/mi^2$	Percent	In.	ft^3/s	$(Mgal/d)/mi^2$	Percent	In.	ft^3/s	$(Mgal/d)/mi^2$	Percent
1969	6.86	29.86	0.327	68.69	6.85	29.82	0.326	68.59	6.61	28.78	0.315	66.20
1970	10.40	45.29	0.495	68.68	10.34	45.04	0.493	68.30	9.92	43.18	0.472	65.48
1971	12.65	55.05	0.602	55.79	12.91	56.22	0.615	56.97	11.99	52.20	0.571	52.89
1972	18.72	81.51	0.891	63.94	18.50	80.55	0.881	63.19	18.08	78.73	0.861	61.76
1973	17.97	78.25	0.856	61.30	18.25	79.47	0.869	62.25	17.98	78.29	0.856	61.33
1974	13.24	57.64	0.630	65.75	13.30	57.91	0.633	66.06	13.18	57.37	0.627	65.44
1975	15.07	65.62	0.718	62.54	14.79	64.38	0.704	61.35	14.34	62.44	0.683	59.51
1976	12.36	53.82	0.589	68.03	12.46	54.24	0.593	68.55	11.99	52.19	0.571	65.97
1977	9.74	42.39	0.464	61.98	9.88	43.00	0.470	62.87	9.40	40.92	0.447	59.82
1978	19.27	83.92	0.918	60.69	19.05	82.95	0.907	59.99	18.64	81.15	0.887	58.69
1979	16.71	72.74	0.795	56.26	16.58	72.20	0.790	55.84	15.93	69.34	0.758	53.63
1980	14.97	65.19	0.713	73.72	14.95	65.10	0.712	73.62	14.57	63.42	0.693	71.71
1981	5.42	23.62	0.258	65.33	5.47	23.81	0.260	65.87	5.29	23.04	0.252	63.73
1982	10.32	44.92	0.491	59.80	10.29	44.79	0.490	59.63	10.05	43.76	0.478	58.25
1983	14.56	63.37	0.693	68.24	14.51	63.16	0.691	68.01	13.89	60.47	0.661	65.12
1984	20.45	89.04	0.974	57.39	20.54	89.41	0.978	57.63	19.54	85.07	0.930	54.84
1985	6.92	30.12	0.329	61.07	6.96	30.30	0.331	61.43	6.66	29.01	0.317	58.82
MEDIAN												
	13.24	57.64	0.630	62.54	13.30	57.91	0.633	62.87	13.18	57.37	0.627	61.33

CALENDAR YEAR BASE FLOW SUMMARY FOR FRENCH CREEK NEAR PHOENIXVILLE, PA

Water	Fixed interval method				Sliding interval method				Local minimum method			
year	In.	ft^3/s	$(Mgal/d)/mi^2$	Percent	In.	ft^3/s	$(Mgal/d)/mi^2$	Percent	In.	ft^3/s	$(Mgal/d)/mi^2$	Percent
1969	6.40	27.88	0.305	67.40	6.43	28.01	0.306	67.71	6.18	26.88	0.294	64.99
1970	11.15	48.55	0.531	66.07	11.17	48.64	0.532	66.19	11.00	47.90	0.524	65.19
1971	14.71	64.03	0.700	59.10	14.81	64.47	0.705	59.51	13.65	59.44	0.650	54.86
1972	18.55	80.76	0.883	59.95	18.54	80.71	0.883	59.91	17.93	78.04	0.853	57.93
1973	16.23	70.65	0.773	60.60	16.41	71.43	0.781	61.27	16.37	71.26	0.779	61.12
1974	12.50	54.44	0.595	70.03	12.54	54.59	0.597	70.22	12.38	53.91	0.589	69.34
1975	16.92	73.66	0.805	64.93	16.58	72.18	0.789	63.63	16.11	70.14	0.767	61.83
1976	11.44	49.79	0.544	67.07	11.53	50.19	0.549	67.60	11.08	48.25	0.528	64.99
1977	10.72	46.68	0.511	56.17	11.01	47.93	0.524	57.67	10.80	47.01	0.514	56.56
1978	18.56	80.82	0.884	63.98	18.22	79.34	0.868	62.81	17.58	76.56	0.837	60.61
1979	18.55	80.78	0.883	58.85	18.42	80.21	0.877	58.44	17.59	76.59	0.837	55.80
1980	11.43	49.75	0.544	73.42	11.41	49.67	0.543	73.31	11.26	49.03	0.536	72.37
1981	5.75	25.02	0.274	66.01	5.80	25.24	0.276	66.57	5.53	24.07	0.263	63.50
1982	10.64	46.34	0.507	59.79	10.61	46.18	0.505	59.58	10.33	44.96	0.492	58.02
1983	16.21	70.58	0.772	60.16	16.24	70.71	0.773	60.27	15.48	67.38	0.737	57.43
1984	19.08	83.05	0.908	62.90	19.10	83.14	0.909	62.97	18.32	79.78	0.872	60.42
MEDIAN												
	13.61	59.24	0.648	63.44	13.67	59.53	0.651	62.89	13.02	56.67	0.620	60.86

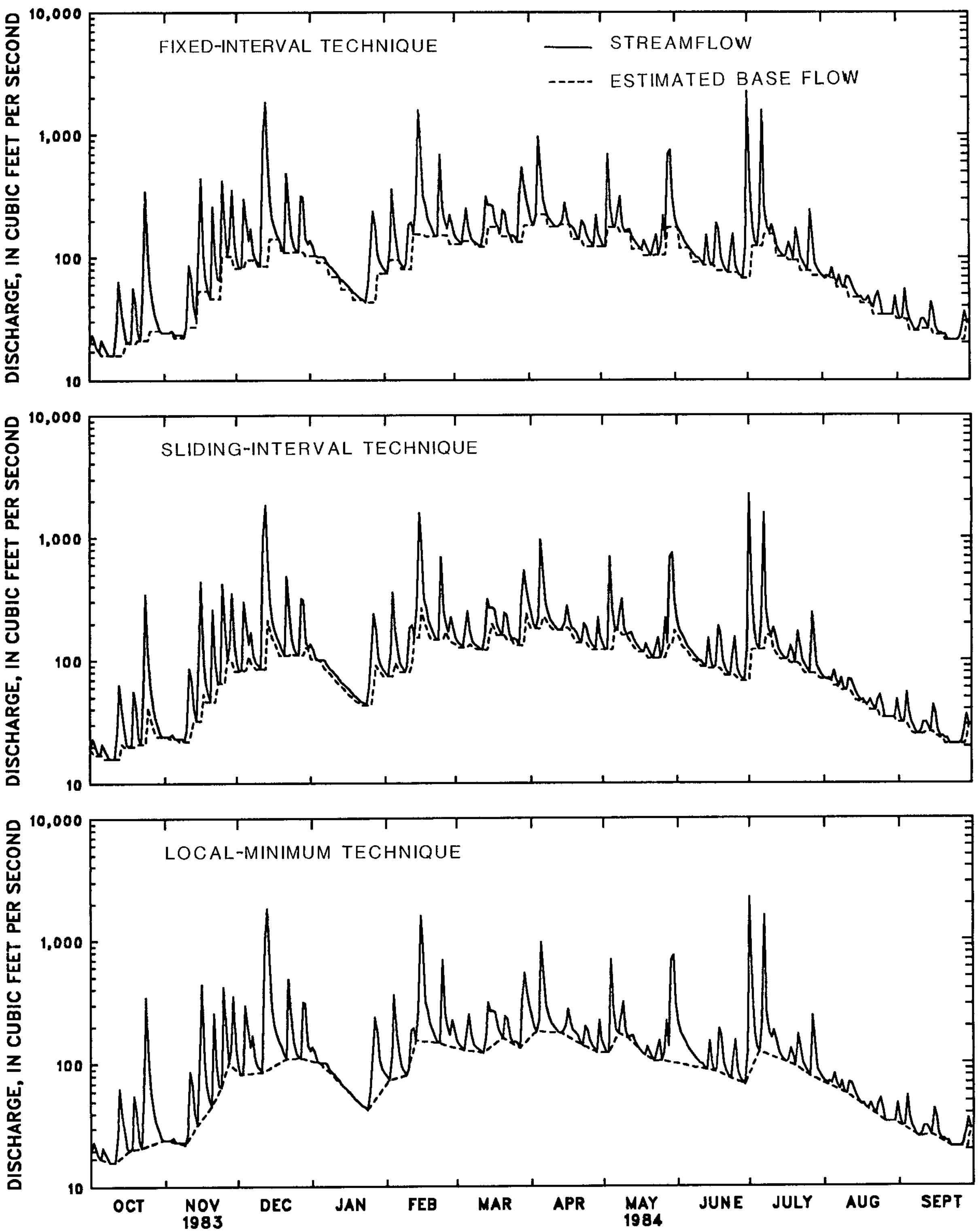


Figure E14.--Daily mean streamflow and base flow for French Creek near Phoenixville, Pennsylvania, 1984 water year.

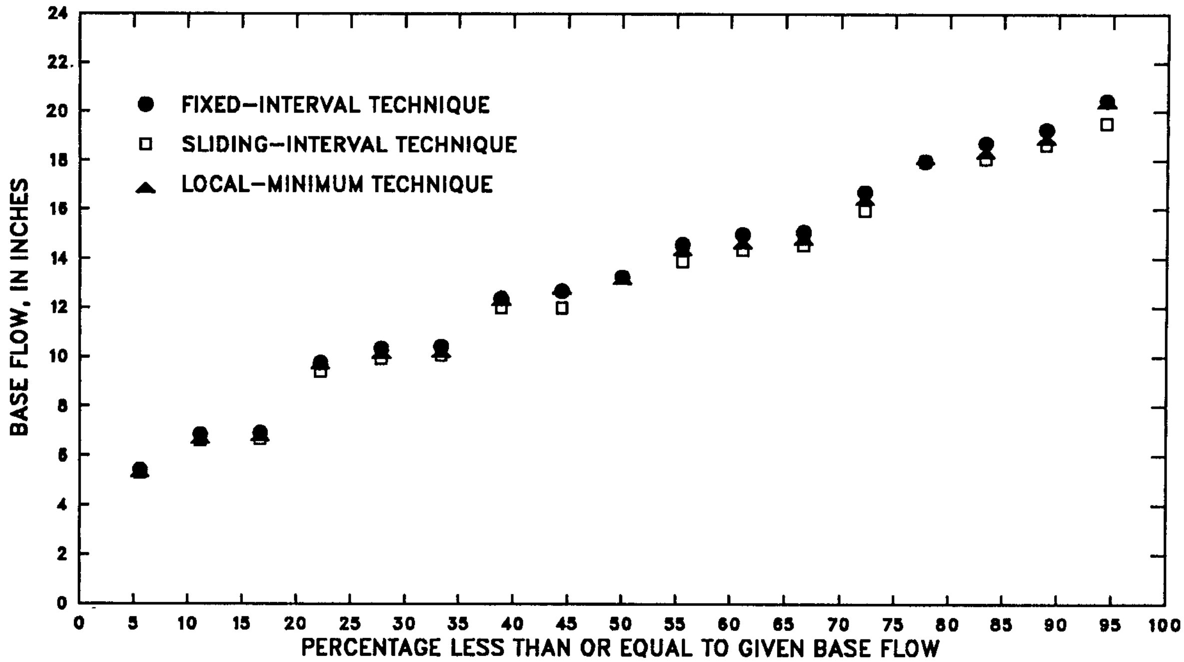


Figure E15.--Frequency distribution of base flow for French Creek near Phoenixville, Pennsylvania, 1969-85.

for three streamflow stations in southeastern Pennsylvania: West Branch Brandywine Creek near Honey Brook (drainage area of 18.7 mi^2), Little Lehigh Creek near Allentown (drainage area of 80.8 mi^2), and Neshaminy Creek near Langhorne (drainage area of 210 mi^2). Base flows are compared in table E5. The sliding-interval technique gave the highest median base flow for two of the three stations, and the fixed-interval technique gave the highest median base flow for one station. The local-minimum technique gave the lowest median base flow for all three stations.

COMPARISON OF COMPUTER- AND MANUAL-SEPARATION TECHNIQUES

Computer hydrograph-separation techniques were compared with those performed manually by base-flow recession and graphical methods for streams in southeastern Pennsylvania. Base flows estimated with computer techniques closely agree with those estimated by manual methods.

A study by Olmsted and Hely (1962) examined the relation between ground water and surface water in the 287 mi^2 Brandywine Creek basin in southeastern Pennsylvania. Olmsted and Hely (1962, p. 4-9) used base-flow recession curves to estimate ground-water contribution to streamflow for 1928-31 and 1952-53. Base flow estimated by Olmsted and Hely (1962) and by computer techiques (fixed interval, sliding interval, and local minimum) is given in table E6. All three computer techniques produced base flows comparable to the manual base-flow recession method. The average base flow for the 6-year period estimated by the sliding-interval technique was 1 percent higher, the estimate by the fixed-interval technique was less than 1 percent higher, and the estimate by the the local-mimimum technique was less than 1 percent lower than the base flow estimated by the manual base-flow recession method. Base flow estimated by computer techniques for individual years ranged from 5 percent lower to 5 percent higher than base flow estimated by the manual base-flow recession method. Base-flow hydrographs for 1952-53 published by Olmsted and Hely (1962, p. 5) are shown for comparison with those estimated by the local-minimum technique in figure E16. The base-flow hydrographs are very similar.

McGreevy and Sloto (1977, p. 38-39) used a graphical hydrograph-separation method to estimate basin ground-water yield in Chester County, Pennsylvania for a dry (1966), near-average (1968), and wet (1973) water year. The average base flow estimated at four streamflow stations by the fixed-interval technique was 6 percent higher, the estimate by the sliding-interval technique was 7 percent higher, and the estimate by the local-minimum technique was 4 percent higher than the base flow estimated by the manual graphical method. Base flow estimated by computer techniques for individual years ranged from 7 percent lower to 19 percent higher than base flow estimated by the manual graphical method.

Biesecker and others (1968, p. 27-29) determined the contribution of ground water to total streamflow for seven stations in the Schuylkill River basin in southeastern Pennsylvania and for Brandywine Creek at Chadds Ford, Pennsylvania for the 1952 and 1964 water years. They used a combination of base-flow recession and curve-fitting methods. The average base flow estimated at the seven streamflow stations by the fixed-interval and sliding-interval techniques was 9 percent lower and the estimate by the local-minimum technique was 14 percent lower than the base flow estimated by the manual method. Base flow estimated by computer techniques for individual years ranged from 7 percent higher to 31 percent lower than base flow estimated by the manual method.

Table E5.--Summary of base flow estimated by computer hydrograph-separation techniques for three streamflow stations in southeastern Pennsylvania, water years 1961-85

[mi^2, square miles]

	Base flow (inches)								
	West Branch Brandywing Creek near Honey Brook (18.7 mi)			Little Lehigh Creek near Allentown (80.8 mi)			Neshaminy Creek near Langhorn (210 mi)		
Water year	Fixed interval	Sliding interval	Local minimum	Fixed interval	Sliding interval	Local minimum	Fixed interval	Sliding interval	Local minimum
1961	12.23	12.58	12.37	11.92	11.93	11.79	12.47	12.54	12.34
1962	9.63	9.89	8.75	8.98	9.21	8.82	6.83	7.29	6.54
1963	7.27	7.18	6.21	9.57	9.40	9.02	5.07	5.09	4.90
1964	9.27	9.07	7.93	7.98	7.99	7.82	7.28	7.27	6.59
1965	6.66	6.57	6.25	5.52	5.50	5.44	5.57	5.22	4.56
1966	7.17	7.19	6.31	5.27	5.23	5.24	4.43	4.24	3.95
1967	8.96	8.77	8.11	7.65	7.68	7.67	11.01	11.11	10.56
1968	9.07	9.05	8.40	10.33	10.31	9.99	7.98	7.95	7.25
1969	7.70	7.76	7.55	7.62	7.59	7.50	6.88	6.59	6.16
1970	9.75	9.77	9.42	10.55	10.54	10.31	9.30	9.17	8.89
1971	10.67	10.83	10.23	15.62	15.69	15.16	10.32	10.18	8.53
1972	14.75	15.11	14.39	21.34	21.41	21.18	14.75	14.30	13.34
1973	14.43	14.70	14.15	23.11	23.29	23.20	14.53	15.03	14.74
1974	11.71	11.68	11.28	17.83	17.94	17.98	11.27	11.45	11.11
1975	15.13	15.23	13.79	19.66	19.60	19.07	13.75	13.52	12.53
1976	14.33	13.76	12.99	18.45	18.61	18.10	8.84	9.21	8.57
1977	11.67	11.85	11.51	12.89	12.93	12.73	5.02	5.15	4.20
1978	16.36	16.77	15.30	20.59	20.64	20.40	13.27	13.08	12.72
1979	15.59	16.20	13.97	16.53	16.64	16.41	13.63	13.74	12.97
1980	14.06	14.16	13.58	14.80	14.84	14.68	11.59	11.49	10.95
1981	6.47	6.52	6.29	6.97	7.00	7.01	6.23	6.21	5.99
1982	8.98	8.67	8.25	10.55	10.48	10.22	8.50	8.44	7.53
1983	14.14	13.92	12.96	14.79	14.71	14.46	11.91	11.91	11.74
1984	17.52	17.40	15.51	23.03	23.02	22.26	14.95	14.54	13.24
1985	9.56	9.64	9.18	9.18	9.21	9.06	4.05	4.07	3.64
Median	10.67	10.83	10.23	11.92	11.93	11.79	9.30	9.21	8.57

Table E6.--Summary of base-flow estimates by manual base-flow recession and computer hydrograph-separation techniques, Brandywine Creek at Chadds Ford, Pennsylvania, 1928-31 and 1952-53

	Base flow (inches)			
Year	Base-flow recession technique[1]	Fixed-interval technique	Sliding-interval technique	Local-minimum technique
1928	18.23	17.63	17.57	17.24
1929	1.73	1.53	1.93	1.99
1930	8.67	8.36	8.40	8.24
1931	6.15	6.31	6.42	6.44
1952	18.68	19.21	19.23	19.01
1953	16.61	17.35	17.46	16.90

[1]From Olmsted and Hely (1962)

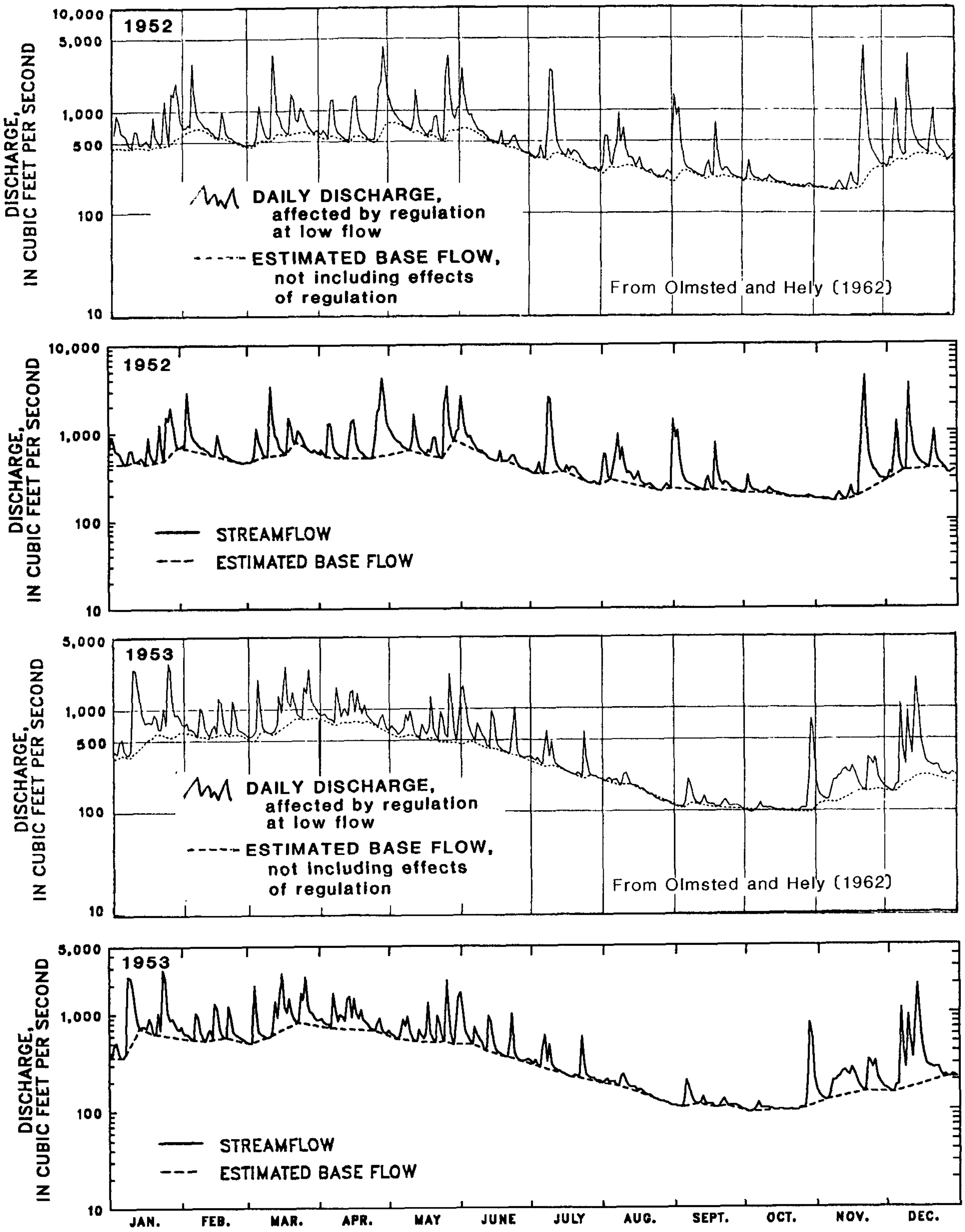


Figure E16.--Daily mean streamflow and base flow for Brandywine Creek at Chadds Ford, Pennsylvania, for 1952-53. Hydrographs from Olmsted and Hely (1962) are based on manual techniques; the other hydrographs are based on computer techniques.

AVAILABILITY OF COMPUTER PROGRAM

The computer code for the hydrograph-separation program and user documentation are available from the Malvern Subdistrict Office of the U. S. Geological Survey, 111 Great Valley Parkway, Malvern, PA 19355. The program length is 2,300 lines of code.

CONCLUSIONS

Computer hydrograph-separation techniques provide a rapid method for estimating ground-water contribution to streamflow. They substantially reduce the time required for hydrograph separation from weeks or days to seconds. Computer techniques give results comparable to manual methods based on base-flow-recession or curve-fitting techniques. They remove subjectivity inherent in manual methods.

REFERENCES CITED

Biesecker, J. E., Lescinsky, J. B., and Wood, C. R., 1968, Water resources of the Schuylkill River basin: Pennsylvania Department of Forests and Waters Bulletin Number 3, 198 p.

Linsley, R. K., Kohler, M. A., and Paulhus, J. L., 1982, Hydrology for engineers, 3rd edition: McGraw-Hill, New York, 508 p.

Olmsted, F. H., and Hely, A. G., 1962, Relation between ground water and surface water in Brandywine Creek basin, Pennsylvania: U.S. Geological Survey Professional Paper 417-A, 21 p.

McGreevy, L. J., and Sloto, R. A., 1977, Ground-water resources of Chester County, Pennsylvania: U.S. Geological Survey Water Resources Investigations Report 77-76, 76 p.

Pettyjohn, W. A., and Henning, Roger, 1979, Preliminary estimate of ground-water recharge rates, related streamflow and water quality in Ohio: Ohio State University Water Resources Center Project Completion Report Number 552, 323 p.

Riggs, H. C., 1963, The base-flow recession curve as an indicator of ground water: International Association of Scientific Hydrology Publication 63, p. 352-363.

Rorabaugh, M. I., 1963, Estimating changes in bank storage and ground-water contribution to streamflow: International Association of Scientific Hydrology Publication 63, p. 432-441.

STATISTICAL AND GRAPHICAL METHODS USED TO DESCRIBE GROUND-WATER QUALITY IN ILLINOIS

R.H. Coupe and K.L. Warner
U.S. Geological Survey, Urbana, Ill.

ABSTRACT

The U.S. Geological Survey uses various statistical and graphical techniques to characterize ground-water quality in Illinois. The statistics and graphics are based on public-supply-well information collected from 1984-87 by the Illinois Environmental Protection Agency and the U.S. Geological Survey. The statistical packages associated with the National Water Information System were used for tabular and graphical representation of ground-water quality in Illinois.

A Geographic Information System was used to display the spatial distribution of wells where selected inorganic constituents in ground water exceeded U.S. Environmental Protection Agency water-quality maximum contaminant levels. For those constituents for which data were sparse but whose presence is important (such as volatile organic carbon, volatile organic aromatic compounds, and pesticides), a Geographic Information System also was used to represent graphically the spatial distribution of the data.

INTRODUCTION

The Illinois Environmental Protection Agency's (IEPA) Division of Public Water Supplies, in cooperation with the U.S. Geological Survey (Survey), sampled approximately 2,080 public-supply wells throughout Illinois during the period 1984-87. The water-quality data collected during this program were useful to IEPA in reviewing current and in proposing new State ground-water-quality standards for health-related chemical compounds.

Prior to proposing the standards, the IEPA first had to analyze incidence concentrations in combination with health effects for a wide range of organic and inorganic constituents. The Survey, using the National Water Information System (NWIS) data bases and a Geographic Information System (GIS), assisted the IEPA in this data analysis.

The need to complete a statistical analysis of these data is immediate but has long-range implications. This is the first analysis of ground-water-quality data from public-supply wells in Illinois. The statistical analysis of that data is useful to IEPA for determining legally enforceable ground-water-quality standards for the State.

This paper documents the steps taken in developing a statistical and graphical description of the quality of ground water in Illinois. The statistical techniques and some of the graphical techniques used are available in NWIS software. Other graphical methods use a GIS available throughout the Survey.

This study was done in cooperation with the Illinois Environmental Protection Agency.

DATA BASE

The data used in this analysis are from one-time-only water samples from Illinois public-supply wells and were collected during March 1984 through July 1987 (Voelker, 1986; Voelker and others, 1988). Approximately 100 wells were selected to be sampled quarterly for 2 years. The first sample only, from each of these sites, was included in the data set. The IEPA plans to sample every public-supply well in Illinois. About 2,080 of 5,000 public-supply wells in Illinois have been sampled as of July 1987. The 2,080 wells that were sampled contain water that is considered to be representative of the five major aquifers in the State.

STATISTICAL METHODS OF ANALYSIS

The IEPA's ground-water-quality data are stored in the Survey's NWIS data base as part of the cooperative efforts between the Survey and the IEPA. The statistical programs used on the NWIS data base were chosen because they provided the needed statistical summaries without having to manipulate the data into some other form for use with a different statistical program.

The simplest of these new statistical programs in NWIS is called QWDETLIMS.F77 (T.L. Schertz, U.S. Geological Survey, written commun., 1988). The output from this program is a listing of all the constituents for selected water-quality records, and the total number of times that each constituent was analyzed (total non-missing values). Also included is a listing of the different detection limits (all values encoded with a '<') for each constituent and a count of the number of times the value from individual analysis equaled or exceeded that detection limit for each year included in the study. An example of the output from the program is shown in figure E17.

This program not only provided a listing of the constituents in the data set, but also indicated those constituents that are present in concentrations above the detection limit and that are suitable for basic statistical analysis. Statistical analysis of a constituent with few values above the detection limit could lead to a misrepresentation of the quality of ground water on a statewide basis. The output from the QWDELTIMS.F77 program indicates if the detection limit for a constituent has changed and when, and gives an indication of when the method of analysis might have changed or a different laboratory was used. The table created by this program allows the user to easily detect inconsistencies in the detection limits and to verify the data before a statistical analysis is done.

To obtain the basic statistics for those constituents that were suitable for basic analysis, a statistical program called QWPRCNTL.F77 was used. The output from QWPRCNTL.F77 (fig. E18) gives the sample size; the maximum, minimum, and mean values; and the percentile ranking of the data, including the median.

```
*******************************************************************************************************
PARAMETER:  LEAD DISSOLVED          01049  TOTAL NON-MISSING VALUES =     25

DETLIM   1984   1985   1986   1987    SUM    % OF TOTAL
  5.00      0      0      0     18     18      72.00
 10.00      0      0      5      0      5      20.00

*******************************************************************************************************
PARAMETER:  LEAD TOTAL              01051  TOTAL NON-MISSING VALUES =   2514

DETLIM   1984   1985   1986   1987    SUM    % OF TOTAL
  5.00    159    410   1013    783   2365      94.07
  1.00      0      0      1      1      2       0.08
  7.00      0      1      0      0      1       0.04

*******************************************************************************************************
PARAMETER:  MANGANESE TOTAL         01055  TOTAL NON-MISSING VALUES =   2557

DETLIM   1984   1985   1986   1987    SUM    % OF TOTAL
  5.00     19     77    155    219    470      18.38
  6.00      0      1      0      0      1       0.04
 10.00      0      0      1      0      1       0.04
 70.00      0      1      0      0      1       0.04

*******************************************************************************************************
PARAMETER:  MANGANESE DISSOLVED     01056  TOTAL NON-MISSING VALUES =     25

DETLIM   1984   1985   1986   1987    SUM    % OF TOTAL
  5.00      0      0      0      2      2       8.00

*******************************************************************************************************
```

Figure E17.--Example output from program QWDETLIMS.F77.

STATISTICAL SUMMARY OF SELECTED WATER QUALITY DATA COLLECTED FROM APR 1984 TO JUL 1987

WATER-QUALITY CONSTITUENT	DESCRIPTIVE STATISTICS: SAMPLE SIZE	MAXIMUM	MINIMUM	MEAN	PERCENT OF SAMPLES IN WHICH VALUES WERE LESS THAN OR EQUAL TO THOSE SHOWN: 95 %	75 %	(MEDIAN) 50 %	25 %	5 %
00610 NITROGEN NH4 ASN TOT	1995	27.000	<0.010	1.042*	3.950	1.050	0.400	<0.100	<0.100
00630 NO2 + NO3 AS N TOT	1998	29.000	<0.100	0.822*	5.350	0.160	<0.100	<0.100	<0.100
00665 PHOSPHORUS TOT AS P	1996	879.000	<0.010	1.262*	0.510	0.100	0.020	<0.010	<0.010
00916 CALCIUM TOTAL EPA	1991	280.000	0.510	80.138	140.000	95.000	75.000	61.000	33.000
00927 MAGNESIUM TOTAL EPA	1991	120.000	0.300	36.955	66.000	45.000	35.000	27.000	15.000
00929 SODIUM TOTAL EPA	1991	1100.000	0.600	58.247	230.000	58.500	28.000	12.000	3.700
00937 POTASSIUM TOTAL EPA	1991	150.000	0.300	3.531	12.000	3.800	2.100	1.300	0.600
00940 CHLORIDE	2006	1700.000	<1.000	41.541*	170.000	38.000	13.000	3.300	<1.000
00945 SULFATE DISS	2005	1500.000	<7.500	103.157*	410.000	120.000	46.000	12.000	<10.000
00951 FLUORIDE TOTAL	1998	4.500	0.100	0.486	1.300	0.600	0.300	0.200	0.100
00956 SILICA, TOTAL	1266	43.000	1.000	14.942	25.000	19.000	15.000	9.800	7.200
01002 ARSENIC TOTAL	2000	96.000	<1.000	4.261*	25.000	2.000	<1.000	<1.000	<1.000
01007 BARIUM TOTAL	1991	23000.000	5.000	228.843	500.000	200.000	90.000	40.000	10.000
01022 BORON TOTAL	1988	2300.000	<8.000	282.523*	1000.000	380.000	150.000	<50.000	<50.000
01037 COBALT TOTAL	1990	30.000	<5.000	3.472*	9.000	<5.000	<5.000	<5.000	<5.000
01042 COPPER TOTAL RECOV.	1990	1700.000	<5.000	9.681*	27.000	<5.000	<5.000	<5.000	<5.000
01045 IRON TOTAL	1990	45000.000	<50.000	1402.073*	4800.000	1800.000	620.000	130.000	<50.000
01055 MANGANESE TOTAL	1990	2100.000	<5.000	89.573*	460.000	74.000	21.000	7.000	<5.000
01067 NICKEL TOTAL	1990	190.000	<3.000	3.362*	12.000	<5.000	<5.000	<5.000	<5.000
01082 STRONTIUM TOTAL	1989	10000.000	5.000	803.072	2900.000	1000.000	390.000	150.000	70.000
01147 SELENIUM TOTAL	1993	39.000	<1.000	0.466*	1.500	<1.000	<1.000	<1.000	<1.000
70300 RESIDUE DIS 180C	1968	3700.000	67.000	571.884	1180.000	648.000	471.000	382.000	300.500

* - VALUE IS ESTIMATED BY USING A LOG-PROBABILITY REGRESSION TO PREDICT THE VALUES OF DATA BELOW THE DETECTION LIMIT

NOTE: MULTIPLE DETECTION LIMITS DURING THE PERIOD OF RECORD MAY RESULT IN VARYING VALUES FLAGGED WITH A "<"

Figure E18.--Example output from program QWPRCNTL.F77.

The rankings created by this program clearly indicated where each individual sample fell in relation to the entire data set. The output was also useful in determining which values needed to be verified before continuing the analysis.

One of the most difficult problems in trying to analyze this type of data statistically is determining what should be done with values that are reported below the detection limit. Values flagged with a less-than sign (<) indicate that the laboratory was not able to quantify the constituent below the reported detection limit; it cannot be assumed, however, that the constituent is not present in quantities below that limit. If these less-than values are used in the analysis, then the risk of bias in the data set can occur. In this situation, the calculated mean would be higher than the true mean; if these less-than values are set to zero (a common solution), the calculated mean will be lower than the true mean. The QWPRCNTL.F77 program offers an intermediate solution. If more than 5 percent of the data are reported below the detection limit, the option to estimate the values below the detection limit is available. The mean and percentile rankings are estimated with a log-probability regression procedure. This method estimates the value below a detection limit and uses those values and the detected values of a constituent to estimate the mean of the data and the percentile rankings. The estimating method was chosen as the best way to handle the problems presented by detection limits in water-quality data. An example of the output from program QWPRCNTL.F77 is shown in figure E19.

GRAPHICAL METHODS OF ANALYSIS

The display of data by graphical methods is a useful technique for presenting information to a general audience. Previously, this information was manually plotted on topographic maps or displayed in tabular format by the IEPA.

Exploratory data analysis (Velleman and Hoaglin, 1981) is a practical method of data analysis that minimizes the use of assumptions. The NWIS software has a very useful procedure for exploratory data analysis--the boxplot.

Boxplots are useful for graphically displaying the statistical distribution of the concentrations of a water-quality constituent. The boxplot (fig. E20) graphically shows where the median of the data lies and of how it relates to the rest of the distribution of the data.

The development of a boxplot starts with the median value. Next, the medians of the data above and below the principle median are found (these are the hinges) and a box is drawn enclosing these hinges (fig. E20). The H-spread is the difference between the hinges and gives the range covered by the middle half of the data. The inner fence is defined as the distance 1.5 times the H-spread from either hinge. A line is drawn to the last data value that is not beyond the inner fence. The outer fence is defined as being three times the H-spread from either hinge. Symbols, *'s, are used to show where data values fall within the outer fence. Values beyond the outer fence are shown with o's and are considered far outside values.

STATISTICAL SUMMARY OF SELECTED WATER QUALITY DATA COLLECTED FROM APR 1984 TO JUL 1987

		DESCRIPTIVE STATISTICS			PERCENT OF SAMPLES IN WHICH VALUES WERE LESS THAN OR EQUAL TO THOSE SHOWN				
WATER-QUALITY CONSTITUENT	SAMPLE SIZE	MAXIMUM	MINIMUM	MEAN	95 %	75 %	(MEDIAN) 50 %	25 %	5 %
00610 NITROGEN NH4 ASN TOT	1995	27.000	0.002	1.042*	3.950*	1.050*	0.400*	0.119*	0.029*
00630 NO2 + NO3 AS N TOT	1998	29.000	0.000	0.822*	5.350*	0.171*	0.045*	0.007*	0.000*
00665 PHOSPHORUS TOT AS P	1996	879.000	0.000	1.262*	0.510*	0.100*	0.020*	0.005*	0.001*
00916 CALCIUM TOTAL EPA	1991	280.000	0.510	80.138	140.000	95.000	75.000	61.000	33.000
00927 MAGNESIUM TOTAL EPA	1991	120.000	0.300	36.955	66.000	45.000	35.000	27.000	15.000
00929 SODIUM TOTAL EPA	1991	1100.000	0.600	58.247	230.000	58.500	28.000	12.000	3.700
00937 POTASSIUM TOTAL EPA	1991	150.000	0.300	3.531	12.000	3.800	2.100	1.300	0.600
00940 CHLORIDE	2006	1700.000	0.040	41.541*	170.000*	38.000*	13.000*	3.300*	0.671*
00945 SULFATE DISS	2005	1500.000	0.377	103.157*	410.000*	120.000*	46.000*	13.677*	3.956*
00951 FLUORIDE TOTAL	1998	4.500	0.100	0.486	1.300	0.600	0.300	0.200	0.100
00956 SILICA, TOTAL	1266	43.000	1.000	14.942	25.000	19.000	15.000	9.800	7.200
01002 ARSENIC TOTAL	2000	96.000	0.000	4.261*	25.000*	2.000*	0.467*	0.100*	0.011*
01007 BARIUM TOTAL	1991	23000.000	5.000	228.843	500.000	200.000	90.000	40.000	10.000
01022 BORON TOTAL	1988	2300.000	2.430	282.523*	1000.000*	380.000*	150.000*	57.674*	18.651*
01037 COBALT TOTAL	1990	30.000	0.214	3.472*	9.000*	4.331*	2.599*	1.560*	0.746*
01042 COPPER TOTAL RECOV.	1990	1700.000	0.001	9.681*	27.000*	3.618*	0.818*	0.185*	0.022*
01045 IRON TOTAL	1990	45000.000	2.708	1402.073*	4800.000*	1800.000*	620.000*	130.000*	36.759*
01055 MANGANESE TOTAL	1990	2100.000	0.062	89.573*	460.000*	74.000*	21.000*	7.000*	1.171*
01067 NICKEL TOTAL	1990	190.000	0.027	3.362*	12.000*	3.619*	1.574*	0.687*	0.207*
01082 STRONTIUM TOTAL	1989	10000.000	5.000	803.072	2900.000	1000.000	390.000	150.000	70.000
01147 SELENIUM TOTAL	1993	39.000	0.004	0.466*	1.500*	0.485*	0.219*	0.099*	0.031*
70300 RESIDUE DIS 180C	1968	3700.000	67.000	571.884	1180.000	648.000	471.000	382.000	300.500

* - VALUE IS ESTIMATED BY USING A LOG-PROBABILITY REGRESSION TO PREDICT THE VALUES OF DATA BELOW THE DETECTION LIMIT

NOTE: MULTIPLE DETECTION LIMITS DURING THE PERIOD OF RECORD MAY RESULT IN VARYING VALUES FLAGGED WITH A "<"

Figure E19.--Example output from program QWPRCNTL.F77 using the option to estimate data points below the detection limit.

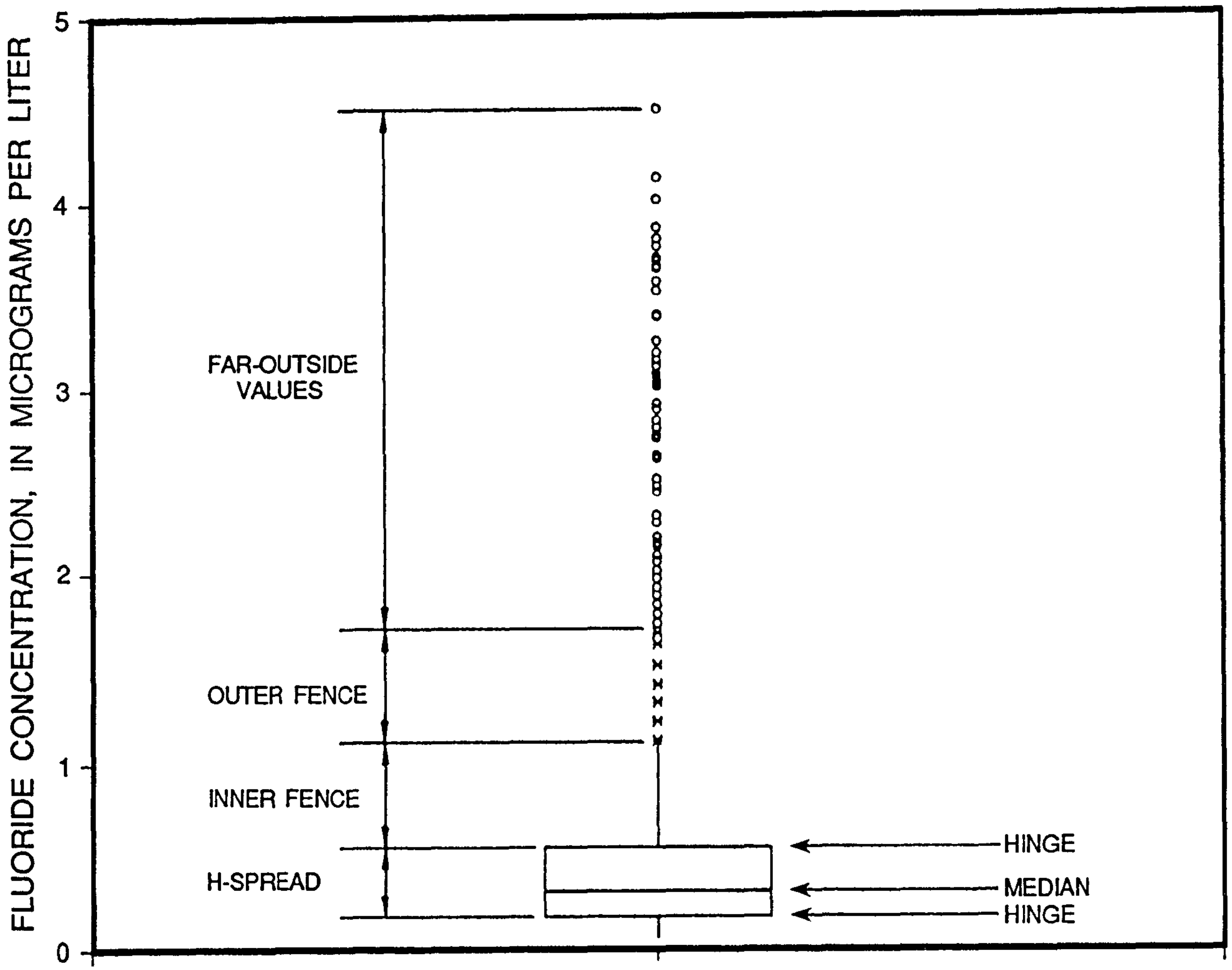


Figure E20.--Example of a boxplot.

One of the investigative uses of boxplots is to organize the data on the basis of hydrogeologic characteristics and to compare the boxplots with one another and with the original data set (fig. E21).

Figure E21 shows that most of the water samples containing arsenic are from wells that have depths less than or equal to 250 feet and that are located in hydrologic units 0512 and 0514 (fig. E21B). The opposite is true for hydrologic unit 0713 where most of the water samples containing arsenic are from wells having depths greater than 250 feet (fig. E21C). These plots by themselves are not conclusive, but their usefulness is in indicating areas where further study is needed.

An ARC/INFO GIS was used to show spatial distribution of wells and detected constituents. This system was chosen because of its data-management capabilities and availability. Most major scientific agencies in Illinois use ARC/INFO; this facilitates the exchange of compatible information.

The chemical constituents were divided into three coverages including (1) inorganic constituents, (2) VOC/VOA (volatile organic carbon and volatile organic aromatic) constituents, and (3) pesticides as suggested by the IEPA. A coverage is a digital analog of a single map sheet stored in a suite of files. The coverages do not include all the constituents sampled, but do include IEPA prioritized inorganic constituents, all VOC/VOA, and detected pesticides (table E7).

Table E7.--Chemical constituents included in each of three coverages: inorganic constituents, volatile organic carbon (VOC) and volatile organic aromatic compounds (VOA), and pesticides

[VOC/VOA includes volatile organic carbon and volatile organic aromatic compounds. This division of constituents was suggested by the Illinois Environmental Protection Agency]

Inorganic constituents	VOC/VOA	Pesticides
Residue upon evaporation	1,1 Dichloroethylene	PCB
	1,1 Dichloroethane	Atrazine
Chlorine	1,2 Trichloroethylene	Alachlor
Sulfate	1,2 Dichloroethane	Metolachlor
Nitrogen	1,1,1 Trichloroethane	
Phosphorus	Carbontetrachloride	
Lead	Trichloroethylene	
Copper	Tetrachloroethylene	
Iron	Chlorobenzene	
	Benzene	
	Toluene	
	Ethylbenzene	

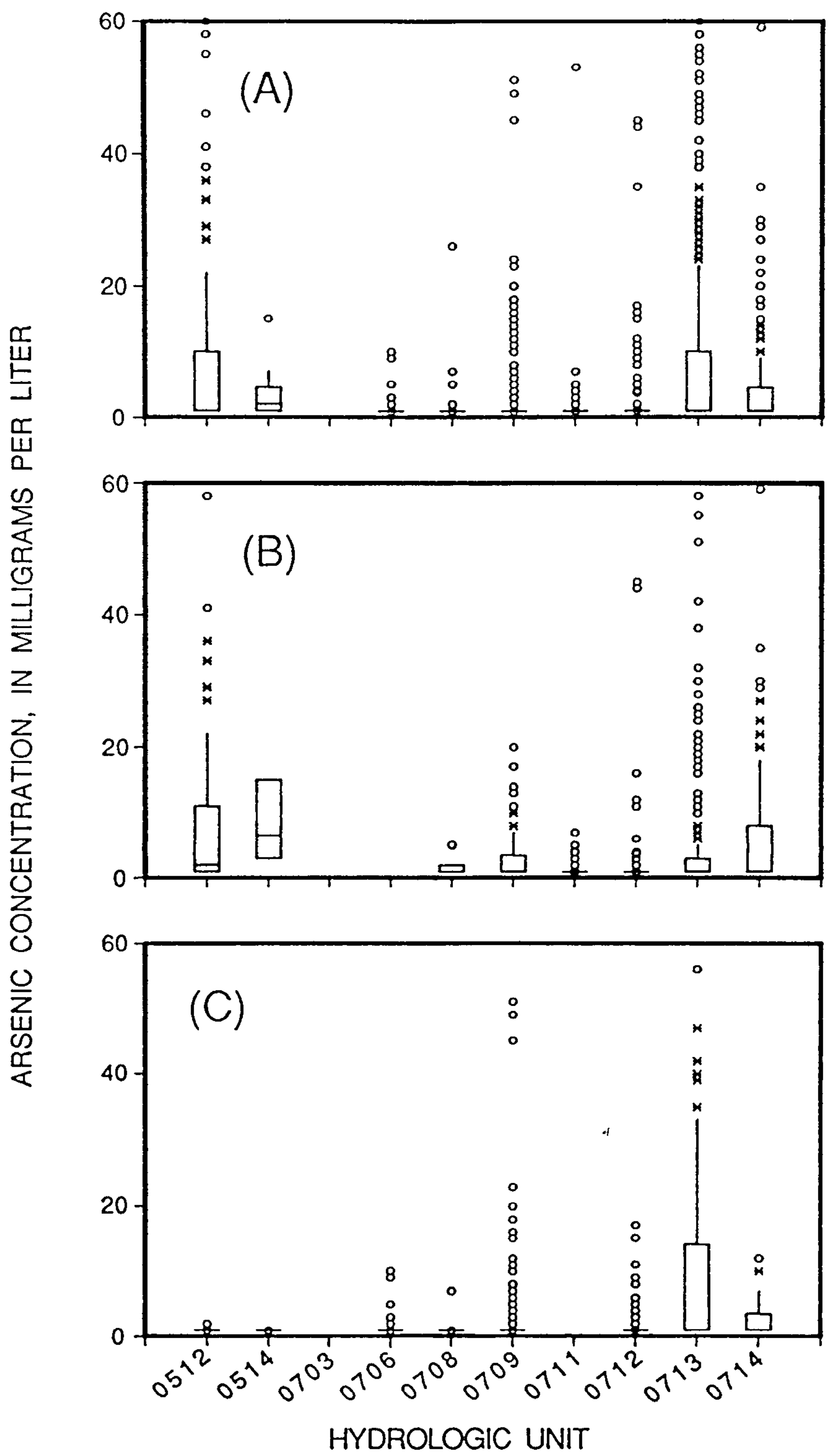


Figure E21.--Arsenic concentration in ground water in Illinois by hydrologic unit for (A) all public wells sampled, (B) wells sampled that were less than or equal to 250 feet deep, and (C) wells sampled that were greater than 250 feet deep.

A GIS was used to display the spatial distribution of wells in which selected inorganic constituents exceeded U.S. Environmental Protection Agency water-quality maximum contaminant levels. For those constituents, for which data are sparse, but whose presence is important (such as volatile organic carbon, volatile organic aromatic compounds, and pesticides), a GIS was used to graphically represent the spatial distribution of the data (fig. E22).

Ground-water sites from NWIS were stored in the data base structure of a GIS. This involved formatting the data to be put into INFO and then into a coverage. A retrieval was run on the NWIS that pulled the first sample for each site as output to a separate file that could be read into INFO. A CPL (command procedure language) formats the file, builds a data base using INFO, and creates a coverage (fig. E22).

A problem arose with encoding concentrations below the detection limit. These values had a preceding less-than sign, which is a character and not an integer. This prevented the values from being defined as integers in INFO. The detection limit value could not be used without the less-than sign because there were some legitimate values at the detection limit. To temporarily solve the problem, so that wells that were sampled but had concentrations below the detection limit could be recorded, all values below the detection limit were defined as '888888'.

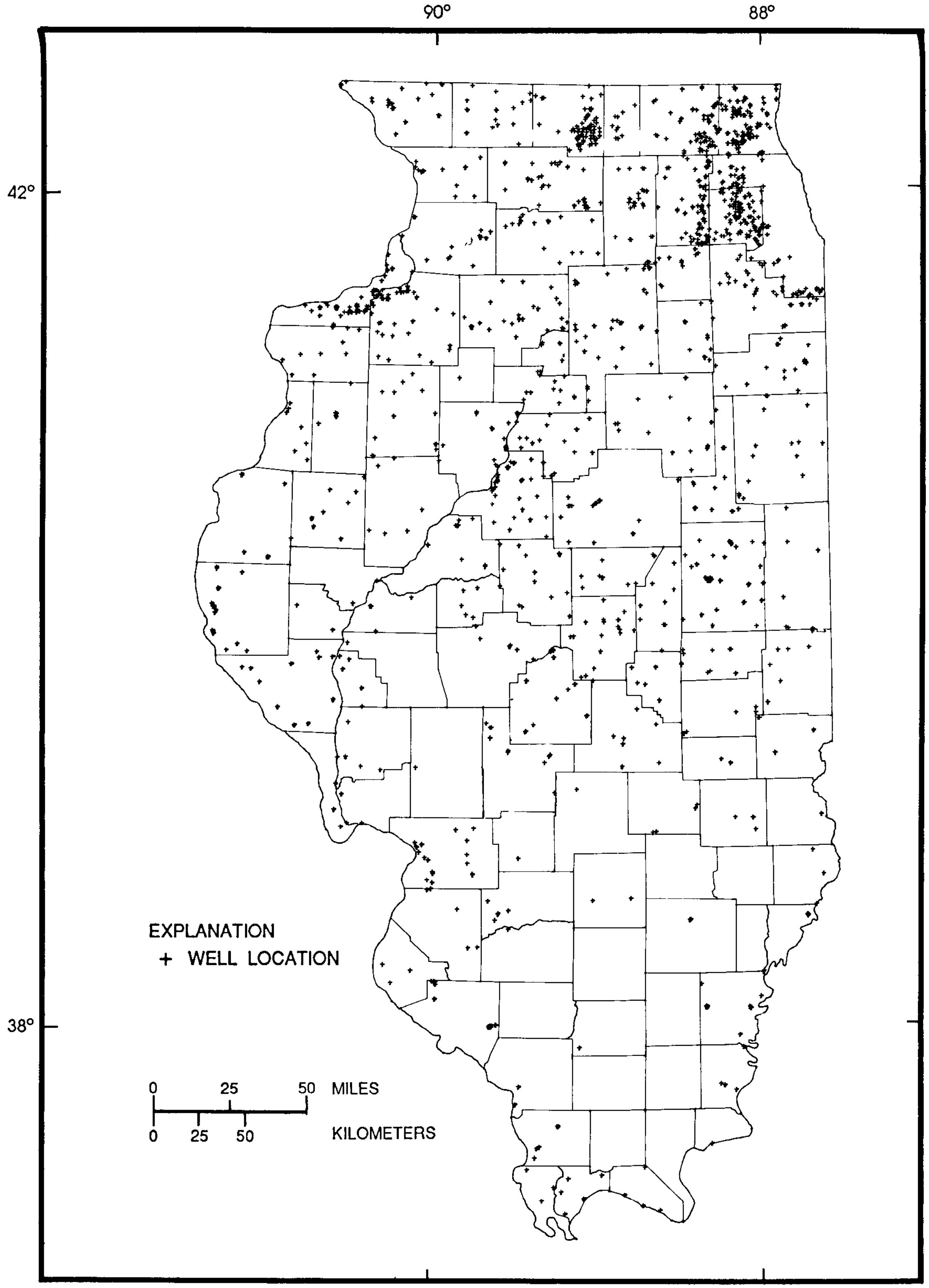
A GIS that uses this type of data has many potential applications. The immediate application was to locate abnormalities in the data. Discrepancies can be detected by overlaying sites for a certain area on a polygon outlining that area. A number of coverages can be used for this purpose including coverages by county, river basin, major aquifer, lake, and river.

By use of the NWIS software in combination with maps developed using the GIS, the IEPA was supplied with the analytical tools they needed to aid in establishing appropriate ground-water-quality standards and to aid in determining the costs of implementing those ground-water standards.

SUMMARY

Illinois has developed water-quality standards for protection of its ground water. The Illinois Environmental Protection Agency and the U.S. Geological Survey sampled 2,080 public-supply wells during 1984-87 for water-quality analysis in Illinois. The U.S. Geological Survey statistically and graphically analyzed this data base.

Chemical constituents for which there were sufficient data to represent ground-water quality in Illinois were identified, and appropriate basic summary statistics were calculated using programs in the NWIS software package. The distributions of these constituents, for selected hydrogeologic characteristics, were represented graphically and compared by use of boxplots. The data were selected and grouped by other hydrogeologic characteristics; these groups were then used to develop boxplots of the data for comparison.



Base from U.S. Geological Survey, 1:2,000,000 digital line graph.

Figure E22.--Location of public-supply wells in Illinois that were sampled as of July 1987.

A Geographic Information System was used to display the spatial distribution of wells in which selected inorganic constituents exceeded U.S. Environmental Protection Agency water-quality maximum contaminant levels. For those constituents for which data were sparse, but whose presence is important (such as volatile organic carbon, volatile organic aromatic compounds and pesticides), the Geographic Information System was used to represent graphically the spatial distribution of the data.

The NWIS software, in combination with maps developed using a Geographic Information System, provided the Illinois Environmental Protection Agency the analytical tools useful for selecting appropriate ground-water-quality standards and for determining the costs of implementing the standards.

REFERENCES CITED

Velleman, P.F., and Hoaglin, D.C., 1981, Applications, basics, and computing of exploratory data analysis: Duxbury Press, Boston, 354 p.

Voelker, D.C., 1986, Observation-well network in Illinois, 1984: U.S. Geological Survey Open-File Report 86-416(W), 108 p.

Voelker, D.C., Oberg, D.J., and Grober, M.J., 1988, Water-quality data from the observation-well network in Illinois, 1985-87: U.S. Geological Survey Open-File Report 87-538, 725 p.

AUTOMATED DATA COLLECTION AND ENTRY TECHNIQUES FOR WATER-USE INFORMATION IN ARKANSAS

By Nancy T. Baker and Terrance W. Holland

U.S. Geological Survey

ABSTRACT

The Arkansas Soil and Water Conservation Commission, in cooperation with the U.S. Geological Survey, annually collects water-use information for more than 25,000 registered wells and surface-water withdrawal sites. To increase the efficiency of collecting and entering large amounts of water-use information, the Survey has developed several automated techniques.

Computer software, designed for use with the Local Applications options of Arkansas' version of the New Site-Specific Water-Use Data System, was developed to automate the printing and distribution of water-use registration forms.

Three methods of using personal computers (located in county Conservation District offices) to improve the efficiency of the data entry process were also developed. One method requires that water-use data entered by each Conservation District office be stored on personal computer diskettes. These data then need to be transferred from the diskette into the New Site-Specific Water-Use Data System data base on the U.S. Geological Survey's minicomputer. The other two methods involve directly accessing the data base by linking personal computers with the use of a modem to the U.S. Geological Survey's computer, which is equipped with a security dial-back system. An advantage of the dial-back system is that costly long distance phone charges can be avoided. It is anticipated that eventually paper copies of the water-use registration form will not be distributed and that all water-use data will be collected and transferred directly to the data base.

INTRODUCTION

A large quantity of potable freshwater is one of Arkansas' most valuable resources. Approximately 5,860 million gallons of freshwater are withdrawn daily from ground- and surface-water sources (Holland, 1987, p. 9). To ensure the protection, development, and management of its water resources, the State has passed legislation requiring registration of diversions of water from streams, lakes, and ponds, and withdrawals of ground water to be registered with the Arkansas Soil and Water Conservation Commission (ASWCC). The ASWCC has been designated as the State agency responsible for water-resources planning, including collection of water-diversion registration information. The U.S. Geological Survey in cooperation with the Arkansas Soil and Water Conservation Commission collects, stores, and manages Arkansas' water-use data for the State. Water-use information for over 25,000 withdrawal sites were reported to the ASWCC during 1986.

The ASWCC entered into its water-use cooperative program with the U.S. Geological Survey in 1985. The State Water Use Data System (SWUDS), available nationwide to U.S. Geological Survey District offices and their cooperators, was the computer program that was initially chosen to manage the data base. Because SWUDS did not meet the specific needs of the ASWCC, the original program (or code) was modified and resulted in a new program called New Site-Specific Water-Use Data System (NEWSWUDS).

Currently (1988), site-specific water-use data are collected by ASWCC by distributing water-diversion registration forms to ground- and surface-water users in the State. Nonagricultural registration forms are mailed directly to the water user. Agricultural forms are distributed through local Conservation District offices in each county. These offices commonly deal closely with the agricultural water users in their district and are able to collect more water-use information than the ASWCC can collect by direct solicitation. In most cases, the water user goes to the Conservation District office and personnel from the office help register their water usage. The Conservation District office sends completed forms back to the ASWCC for entry into NEWSWUDS. Because the large volume of site-specific data reported annually to the ASWCC has become increasingly difficult to collect and process effectively, the U.S. Geological Survey has developed several techniques to automate data collection and entry. This paper describes the techniques that have been developed by the Arkansas District of the U.S. Geological Survey to automate water-use data collection and entry.

AUTOMATED DATA COLLECTION

Applications software was added to the Local Applications system in NEWSWUDS to streamline the data-collection process. The Local Applications system is a feature of NEWSWUDS that allows locally developed applications to be installed in the NEWSWUDS system by using Fortran "envelope" routines and templates that are provided as a part of the system.

The ASWCC Forms Routine is one of the local application software packages installed in NEWSWUDS in Arkansas. The ASWCC Forms Routine prints water-diversion registration forms for ground- and surface-water withdrawal sites (figs. E23 and E24). The routine can print blank registration forms without water-user information or it can preprint pertinent water-user information, such as water-user identification number, name, address, well and pump data, and location of diversion. Preprinting the forms assures that water users who submitted forms the previous year will be sent a form with current information that may be updated as necessary. The amount of time needed to complete the form is reduced considerably, and generally, the only new information that is required is the annual water-use data.

To create the preprinted forms with the ASWCC Forms Routine, the user needs to first retrieve the appropriate information from the NEWSWUDS data base. A "Command Procedure Language" (CPL) program, which operates at the system level, has been set up to retrieve the data from NEWSWUDS. The CPL allows the user to retrieve information quickly and easily, and in the order required by the ASWCC Forms Routine. The CPL also eliminates the typing of a long sequence of commands each time a retrieval is made. The retrieval can be submitted as a batch job; this frees the the system for other uses, and allows

WATER WITHDRAWAL REGISTRATION FORM
AGRICULTURE

1. Name of Landowner ____________
2. Water User ID. ____________
3. Use of WATER:(check one)
   ____AG Agriculture (livestock, fish farms) ____IR Irrigation
4. SIC Code 1st ____________. 2nd ____________.
5. ASWCC Permit # ____________ 6. Telephone number ____________
7. Name of Diverter ____________
8. Hydrologic Unit Code ____________
9. Street Address ____________
10. City name ____________ 11. ZIP Code ____________
12. FIPS County code ____________ 13. FIPS State Code ____________
14. State _____ 15. Action Code _____ 16. Source type. ____GW ____SW
17. Conservation District Permit #: Topographic Quad # ___ ___ ___ ___
   Operator # ___ ___ ___ WELL # ___ ___
18. Aquifer (GW only) ____________
19. Reclaimed waste (GW only) _____ YES _____ NO
20. Name of lake, ditch, bayou, or tributary (SW only) ____________
21. Was a dam or other obstruction of streamflow utilized (SW only)?
   _____ YES _____ NO
22. Method of Irrigation: ____Flood, ____Furrow, ____Sprinkler, ____Other
23. Diameter of pipe ____________ 24. Depth of well ____________
25. Location of diversion or withdrawal point (Legal description):
   _____ 1/4 of _____ 1/4, Section ____, Township ____, Range ____
26. Location of withdrawal: Latitude ____________ Longitude ____________
27. Method of Diversion:
   ____Stationary pump ____Portable Pump ____Gravity
28. Type of power ____ELC ____LPG ____Diesel ____Other
29. Pump Horsepower ____________
30. Year 19____
31. Annual amount withdrawn(acre ft.) 1987 ____________ 1988 ____________
32. Measuring Method: (circle one)
   __E__ Estimated __M__ Measured __C__ Calculated.
33. Restrictions: (circle one) __Y__ __N__
34. Salinity: (circle one) __F__Fresh __S__ Saline __U__ Unknown
35. Estimate total withdrawal per month.

	1987	1988		1987	1988		1987	1988
JAN			FEB			MAR		
APR			MAY			JUN		
JUL			AUG			SEP		
OCT			NOV			DEC		

36. (a) Kind of crop	(b) Number of irrigated acres	(c) Amount of water applied (ft)	(d) Total acre-ft. (b x c)

____________ (Signature) ____________ (Date) ____________ (County of Diversion)

Figure E23.--Example of Arkansas Soil and Water Conservation Commission water-withdrawal registration form for agricultural water users.

WATER-USE REGISTRATION FORM

1. Name of Facility ______

2. Water-user Id. # ______

3. Use of water: (check one)

WS - Public Supply Use	____	PF - Fossil Fuel Power	____
CO - Commercial Use	____	PG - Geothermal Power	____
IN - Industrial Use	____	PN - Nuclear Energy Power	____
MI - Mining	____	PH - Hydroelectric Power	____

4. SIC Code (1)______ (2)______ (3)______

5. Permit# or PWS#______ 6. Telephone Number (___)______

7. Location of Facility: Latitude______ Longitude______

8. Hydrologic Unit Code______

9. Address ______

10. City ______ 11. ZIP Code ______

********* COMPLETE 12-18 FOR PUBLIC WATER SUPPLY FACILITIES ONLY *******

12. Total Ground & Surface Water withdrawals(MILLION GALS.)______

13. Water Purchased From Other Facilities (Amount in MILLION GALS.):

Facility Name ______ Amount:______
Facility Name ______ Amount:______

14. Water Sold To Other Facilities (Amount in MILLION GALS.):

Facility Name ______ Amount:______
Facility Name ______ Amount:______

15. Water Used For Facility Maintenance (backflushing, plant operations, losses due to seepage and leaks - MILLION GALS.)______

16. Domestic Population Served ______

17 Deliveries to Users (Do not include water sold to other facilities):

Water Furnished To:	Total Water Delivered	Number of Connections
Domestic Households	______	______
Commercial	______	______
Industrial	______	______
Mining	______	______
Agriculture	______	______
Irrigation	______	______

18. Current Rate Structure:

______ ______ ______ ______ ______

______ (Name of Plant Manager) ______ (Date) ______ (County of Diversion)

FILL OUT A SEPARATE PAGE FOR EACH WELL OR SURFACE WATER DIVERSION POINT

19. Measurement Point ID______

20. Local Descriptive ID______

*** If CENTROID, amount of water withdrawn (#29-30) represents the sum ***
*** of water withdrawn from all withdrawal points for this facility. ***

21. Action Code______ (If Action Code is DL skip to #29)

22. Source of Water (Ground, Surface, or Transfer) ______

23. If Surface Water withdrawal:

A. Name of Lake or Stream:______

24. If Ground Water withdrawal:

A. Name of Aquifer: ______

B. Depth of Well ______

C. Name of Driller ______

25. Pump Information:

A. Pump Capacity (Horsepower) ______

B. Type of Power (check one): Electric ____ LPGas ____ Diesel ____ Other ____

C. Type of Pump (check one): Stationary ____ Portable ____ Gravity ____ Other ____

D. Diameter of Well or Intake Pipe: ______

26. Location of Well or Withdrawal Point (Well or Intake Structure):

Latitude ______ Longitude ______

27. Hydrologic Unit Code ______

28. Withdrawal Amounts Obtained From (check one):

Billing Records____ Pump Capacity + Hours Operation____
Meter Readings ____ Other (list)______

**** For questions 29-30, withdrawal values reported in (check one):****

Gallons____ Thousand Gallons____ Million Gallons____

29. Total Water Withdrawn From This Withdrawal Point (nn):______

30. Amount of Water Withdrawn From This Withdrawal Point (nn):

nn	nn	nn
JAN______	FEB______	MAR______
APR______	MAY______	JUN______
JUL______	AUG______	SEP______
OCT______	NOV______	DEC______

Figure E24.--Example of Arkansas Soil and Water Conservation Commission water-withdrawal registration form for nonagriculture water users.

users not familiar with the NEWSWUDS system to retrieve data for the printing of registration forms.

After the appropriate information is retrieved from NEWSWUDS, the ASWCC Forms Routine is selected from the Local Applications menu. The user responds to the queries in the routine and the water-user information retrieved earlier is automatically printed on the water-diversion registration forms. To improve distribution, the forms are retrieved and printed according to (1) the type of water use (agriculture (fig. E23) or nonagriculture (fig. E24)), (2) the source type (ground or surface water), and (3) the county in which the withdrawal site is located.

Another Local Applications feature added to NEWSWUDS is the ASWCC Mail Labels Routine. This routine generates mailing labels from the data retrieved for the ASWCC Forms Routine. Labels are printed in the same order that the forms are printed. In Arkansas, mail labels are generated for nonagricultural water users.

AUTOMATED DATA ENTRY

Although automation of registration form processing and distribution has helped to streamline the data collection process, the ASWCC does not have the personnel necessary to process effectively all the registration forms that are received. Most water-diversion registration forms received by the ASWCC are from the agricultural region in eastern Arkansas. If the ASWCC could divide the task of entering the water-use data collected in the large usage counties of eastern Arkansas among the local Conservation District offices, data entry time could be greatly reduced. Techniques to divide data entry among ASWCC and Conservation District personnel are being tested in a few Arkansas counties. Many of the Conservation District offices are being equipped with personal computers (PC's) and several options using the computers for remote data entry are being investigated by the USGS and the ASWCC.

Personal Computer Diskette Method

The personal computer diskette method requires that each Conservation District office enter water-use data on PC diskettes. Data are entered using Keydisk, which is a software package that formats water-user information for entry into NEWSWUDS. Keydisk is part of NEWSWUDS, however, a PC version of Keydisk is available from the USGS. The data entered into the PC version of Keydisk would be saved on a diskette. The diskettes would then be sent to the ASWCC and loaded into the NEWSWUDS data base. The advantages of this system are: (1) the Conservation District offices do not need to purchase any additional equipment (phone lines or modems), (2) Conservation District personnel could enter the data at their convenience and would not be restricted to a schedule of data entry, which is a possibility for the other methods, and (3) there would not be a slowdown of the U.S. Geological Survey's computer because of increased usage load. The disadvantages include: (1) multiple handling of the diskettes, (2) an extra step is involved in processing the information (the diskettes would be sent to the ASWCC for loading), and (3) there would be several copies (1 for each of the 26 counties in eastern Arkansas) of the Keydisk software to maintain.

Dial-Back Method

The dial-back method requires accessing the NEWSWUDS data base directly by linking the PC's by modem to the U.S. Geological Survey's computer, which is equipped with a security dial-back system. The dial-back system consists of placing a modem at the PC end of the system and a dial-back controller at the U.S. Geological Survey's computer end. To use the system, the person at the PC has to call the dial-back controller from a touch-tone phone, enter an access code, and wait for the prompt to hangup the phone. The access code has to be preprogrammed in the dial-back controller. Upon receiving the code, the controller calls the number in its library for that access code and completes the connection. After the system is connected, a signal is sent to the user to set the terminal baud rate. The computer system in the Arkansas District office of the Geological Survey has baud rate options of 300, 1200, and 2400 baud. Upon setting the baud rate, the user can login to the U.S. Geological Survey's computer system.

There are several advantages to the dial-back system that include: (1) Conservation Districts will not incur costly long distance phone charges, (2) Conservation District personnel would have access to the NEWSWUDS data base and would be able to make their own retrievals, and (3) there would only be one copy of Keydisk to maintain. Eventually, paper copies of the water-diversion registration form might not be needed, and water-use data can be collected and directly transferred to the NEWSWUDS data base.

Disadvantages include the possibility of having to enter data on a restricted schedule. Presently (1988) the computer system in the Arkansas District of the U.S. Geological Survey does not have enough dialup lines to handle 26 additional users accessing the NEWSWUDS software. A schedule may have to be set up to assure that only a few counties are entering data at a given time. Another disadvantage is that many of the counties might have to install an additional phone line and might have to purchase a modem.

Combination Method

The combination method involves using the dial-back system to transfer data from the PC to the U.S. Geological Survey's computer, and using the PC version of Keydisk to enter the data. Data would be entered at the PC without linking to the U.S. Geological Survey's computer. These data would be stored in the PC until all data for a particular county are entered. Conservation District personnel would then transfer all of the data at one time to the U.S. Geological Survey's computer system by using the dial-back controller. Conservation District personnel would still have access to information in the NEWSWUDS data base, and this method would not cause a heavy usage load on the U.S. Geological Survey's computer. Although this method would reduce the multiple handling of PC diskettes, there would still be several copies of the PC version of Keydisk to maintain.

CONCLUSION

The development of techniques to automate data collection and entry should greatly increase the efficiency of data management in Arkansas. By increasing the efficiency of water-use data management, more attention can be given to improving the quality of the data collected. Automation of data collection assures a more thorough distribution and collection of water-use information. Automation of data entry divides the task of entering large quantities of data among Conservation District personnel in each county and reduces the workload of the Arkansas Soil and Water Conservation Commission. Of the three automated methods discussed, no single method will likely be appropriate for every county in eastern Arkansas. The method appropriate for an individual county will probably be dependent on the facilities and needs of the Conservation District office.

REFERENCES

Holland, T.W., 1987, Use of water in Arkansas, 1985: Arkansas Geological Commission Water-Resources Summary 16, 27 p.

AUTOMATION OF THE WATER-USE DATA BASE FOR MINNESOTA

By Lee C. Trotta

U.S. Geological Survey

Water Resources Division

St. Paul, Minnesota

ABSTRACT

Because of the complexity and scope of water use, management of water resources requires classification and automation of these data. Data are collected for 10 different categories of water use for the U.S. Geological Survey's National Water-Use Information Program. Water-use managers in Minnesota needed a system to satisfy complex management needs that would identify users in a specific location, describe the source of water, type of use, and rates and volumes of water used. Summaries of water use within a county or watershed were required for transfer of these data to the U.S. Geological Survey's Aggregated Water-Use Data System. A computer based relational data-base-management system was chosen and developed jointly by the Minnesota Department of Natural Resources and U.S. Geological Survey to automate data-base management of water for the State and national programs. Data aggregation programs, a simulation model based on Standard Industrial Classification Codes to determine the economic value of water, and various utility programs have been developed. Many decisions by Federal, State, and local water-resources managers for mandating water-use restriction, initiating ground-water studies, and developing local comprehensive water plans are streamlined by automation of the water-use data base.

INTRODUCTION

Everyone uses water and water is used in many different ways. Water is withdrawn from sources on the surface or in the ground, used for one or more reasons, and then consumed or discharged back to the ground or surface. The U.S. Geological Survey's National Water-Use Information Program currently (1983) collects data on withdrawal, use, and disposition for 10 different categories of water use. These categories are thermoelectric power generation, hydroelectric power generation, irrigation, livestock, commercial, domestic, industrial, mining, public supply, and sewage treatment.

Users of large amounts of water may affect the quantity and quality of water available to other users. The State of Minnesota monitors water use for its large users. Water-use information is collected by a variety of State agencies to satisfy various legislative mandates. Because of the complexity and scope of water use, management of the water resource requires classification of water-use data and automation of data entry, processing and retrieval.

This report describes the Minnesota Water-Use Data System (MWUDS), computer methods used to automate the system, and programs to use the data base. MWUDS was developed independently from the State Water-Use Data System, distributed by the U.S. Geological Survey (Survey).

Permitting Systems in Minnesota

Permitting systems are used to obtain site-specific water-use information in Minnesota. The State of Minnesota has established one permit system for water appropriations greater than 10,000 gallons per day (or 1 million gallons per year, whichever is less) and another to monitor the quality of water dis-charged. Water appropriations (withdrawals) are reported to the Minnesota Department of Natural Resources (MDNR). Rates of compliance with the reporting provision of permits is about 85 percent among irrigation users and 95 percent among all other users (Young, 1987, p. 6).

The MDNR collects information from permit reports in 6 of the 10 water-use categories defined by the Survey. However, the MDNR has defined about 50 subcategories to allow more detailed analysis (table E-8). The MDNR also collects data on aquaculture (code 72), quality improvement (codes 60, 61 and 71), and preservation (codes 60, 61, and 74). These categories are not part of the current focus of the Survey. Relations among water-use categories specified by both MDNR and the Survey were defined by Marilee Horn (1984, p.4) and are revised and expanded upon in the second part of table E-8. Some Survey-specified categories in this part of the table may contain only limited data (hydroelectric power generation and sewage treatment), or be completely absent (livestock and domestic) because of the nature of MDNR's withdrawal-permit reporting requirements. Withdrawals for three of these Survey-specified categories (hydroelectric power generation, livestock, and domestic) have been and are estimated by the Survey annually since 1984, and on a 5-year schedule before that (Trotta, 1988). A water-quality-based permit system regulated by the Minnesota Pollution Control Agency (MPCA) provides some data on water returned to the environment at sewage treatment plants.

COMPUTER METHODS

Thirty-nine years passed between issuance of the first Minnesota water-use permit and development of a rudimentary data-storage system in 1976. During that time, information needed by State planners was retrieved by leafing through thousands of pages of data in notebooks (Gil Young, Minnesota Department of Natural Resources, written commun., 1987).

In 1982, the Minnesota Department of Natural Resources and the U.S. Geological Survey began a joint project to implement a water-use data-base-management system (Horn, 1986). After a long search for applicable computer-based systems, a computer based relational data-base-management system was chosen (fig. E-25) that allows staff with minimal training or computer experience to input, update, and retrieve data. The Land Management Infor-

Table E-8.--Water-use codes and categories in the Minnesota Water-Use Data System

Code	Category used by the Minnesota Department of Natural Resources	Code	Category used by the Minnesota Department of Natural Resources
3	Well abandoned	43	Mine processing (other than sand and gravel washing)
4	Terminated		
5	Active not under permit	44	Sand and gravel washing
		45	Sewage treatment
10	Waterworks	46	Petroleum (chemical processing)
11	Municipal	47	Metal processing
12	Private waterworks (Trailer courts/ small housing units/and so forth)	48	Non-metallic products (rubber/plastic/glass)
13	Commercial and institutional	50	Temporary
14	Cooperative waterworks	51	Construction (non-dewatering)
15	Fire protection	52	Construction (dewatering)
16	State parks/waysides/highway rest areas	53	Pipeline and tank testing
		54	Landscape watering
20	Power generation	60	Water level maintenance
21	Hydropower	61	Basin (lake) level
22	Steam power cooling (once through)	62	Mine dewatering
		63	Quarry dewatering
23	Steam power cooling (wet tower)	64	Sand/gravel pit dewatering
		65	Tile drainage and pumped sumps
24	Steam power cooling (ponds)		
25	Steam power (other than cooling)	71	Pollution confinement
		72	Hatcheries and fisheries
26	Nuclear power	73	Snow making
		74	Peat fire control
30	Air conditioning		
31	Commercial building	80	Non-crop irrigation
32	Institutions (schools, hospitals)	81	Golf course
		82	Cemetery
33	Heat pumps	83	Landscaping
34	Coolant pumps	84	Sod
35	District heating	85	Nursery
		86	Orchard
40	Industrial processing		
41	Food and livestock (agricultural processing)	90	Major crop irrigation
		96	Wild rice irrigation
42	Paper/Pulp		

Codes	Category used by the U.S. Geological Survey
20, 22-26	Thermoelectric power generation
21	Hydroelectric power generation
54, 80-86, 90, 96	Irrigation
30-35, 73	Commercial
40-42, 46-48, 50-52	Industrial
43, 44, 62-64	Mining
10-16, 53	Public supply
45	Sewage treatment

mation Center (LMIC), part of Minnesota's State Planning Agency, helped design the data-base structure and programs needed. Menu-driven programs provide users access to water-use data for approximately 11,500 withdrawal sites currently (1988) in the system.

Data storage was not the only purpose for Minnesota's water-use program. Water-use managers needed a system that would identify users in a specific location in the State, describe the source of water (such as well, lake, or stream), the type of use, and the rate of use. Summaries of water use within a county, a watershed, or statewide were needed for input into the Survey's Aggregated Water-Use Data System (AWUDS).

One of the first accomplishments of MWUDS automation was the development of data-aggregation programs (Land Management Information Center, 1984; Trotta, 1988), which are useful for calculating county and watershed water-use summaries. After verification, this aggregate information can be automatically transferred to the Survey's Aggregated Water-Use Data System (AWUDS). These data summaries are useful for analyzing regional water use trends.

Each of the water-use categories corresponds to a group of related Standard Industrial Classification (SIC) codes (Snavely, 1986, table 5). SIC codes define and classify the entire field of economic activities (Office of Management and Budget, 1972). These SIC codes are included on tax forms used by businesses and on surveys made by the U.S. Department of Commerce, Bureau of the Census. Individuals or facilities are thus categorized on the basis of the purpose of their water use in relation to their economic activity. Published surveys of use ratios, such as that by Debra Snavely (1986, table 7A), when combined with economic reports, such as those published by the Minnesota Department of Jobs and Training (1986), allow economic-policy-analysis simulation models to estimate water-use amounts for each SIC code (Young, 1987, p. 5-9). These estimates are used when actual data are not available.

Combining information in this way from several data bases depends on a linkage or relation between them. When one data base has an item in common format with another, they are termed "RELATED."

Sometimes stand-alone data bases, such as Well Log Listing System (WELLS) or Permit Information Exchange (PIX), are related to MWUDS see figure E-25). Well Log Listing System contains a brief description (location, aquifer and well-construction data) for approximately 100,000 well logs. Locational search routines are available to verify or transfer aquifer data about a particular user's source of water. Menu-driven systems recently have been developed. Well Log Listing System also has a routine to load data to a geographic information system. Permit Information Exchange contains detailed information from well permits on fees and legal actions for the regional water-data network. The permit-handling software in PIX does the regular name and address maintenance on permit applications. Well Log Listing and PIX are linked with either a common geographic-location code or reference numbers that help identify the same water user in different data bases (Horn, 1986, p. 11). The program

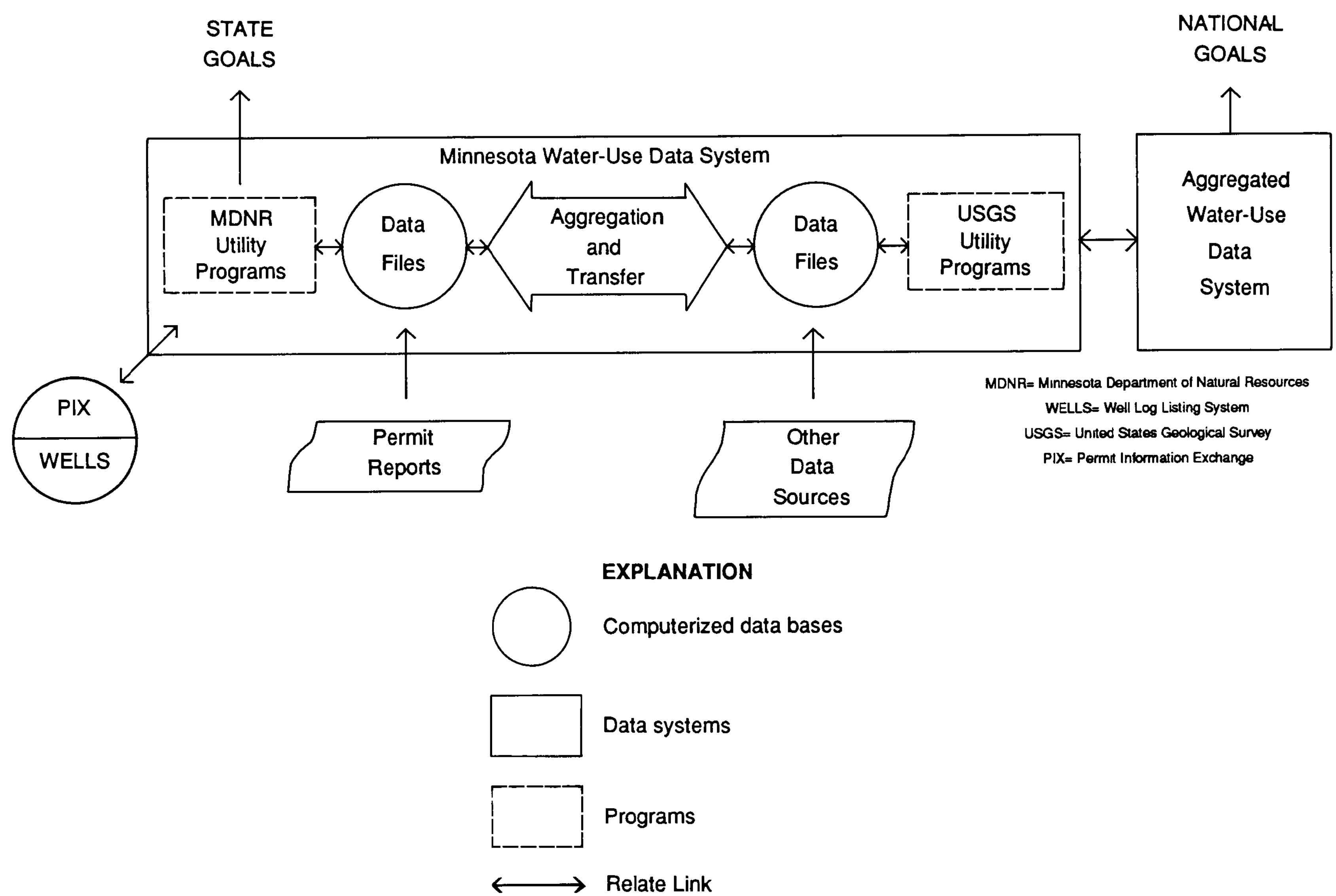


Figure E25.--Computer methods used in the Minnesota Water-Use Data System.

PERMSRCH allows easy access to both historic and active data in a flexible and user-friendly way. A partially menu-driven system called MAILLIST uses information from PIX and MWUDS to produce written communications and forms for reporting pumpage data (Richard Gelbmann, Minnesota Department of Natural Resources, written commun., 1988). Computer linkage to other data bases is described by Horn (1986, p. 18-25).

Utility programs have been developed by MDNR to provide data for many specific applications. Some of the applications are to (1) aid in tracking permits once they have been issued, (2) ensure compliance with regulations or restrictions, and (3) assist the State's education functions with illustrative reports. The MDNR program INDEX, for example, is useful for producing reports that summarize 3 years of site-specific data for any MDNR administrative region (Richard Gelbmann, Minnesota Department of Natural Resources, oral commun., 1988). Search routines in MWUDS can quickly outline, for example, areas where water-level declines may be related to municipal lawn sprinkling or irrigation. The MDNR also has developed input programs for an economic model that determines the economic value of Minnesota water for various water-use scenarios. The economic value of conservation initiatives and reprioritization of use is discussed in a report that summarizes resultsof this model (Adelsman and Bloomgren, 1987, p. 19-22).

U.S. Geological Survey utility programs have been developed that provide information for trend analysis. The AWUDS software, distributed by the National Water-Use Information Program, accepts aggregated data from MWUDS and produces graphical output, pie diagrams, and choropleth maps of the State, depicting county or watershed data. The program QA2, part of the AWUDS, creates a table useful for comparison of water-use data for 1980 (or another reliable base year) and any other data year available (Howard A. Perlman, U.S. Geological Survey, written commun., 1987). Additional programs were developed in the Minnesota District of the U.S. Geological Survey to extract data needed by the national program from specialized data files associated with MWUDS. These programs calculate municipal population by watershed, sewage-treatment facilities by watershed, water withdrawals for livestock by watershed, and diversion of water across watershed boundaries. Predictions of the effect of diversions or changes in pumpage may be made within a short time frame.

SUMMARY

Automation improves access to data and efficiency of data-analysis capabilities. Careful documentation of procedures used in data-search and reporting commonly points out inefficient manual steps correctable by automation. Many management decisions for mandating restrictions, initiating ground-water studies, and developing local comprehensive water plans are streamlined by automation of the water-use data base.

REFERENCES CITED

Adelsman, H.R. and Bloomgren, P.A., 1987, Summary--The economic value of water: Department of Natural Resources, Division of Waters Report to the Legislative Commission on Minnesota Resources, St. Paul, 22 p.

Horn, M.A., 1984, Annual ground-water use in the Twin Cities Metropolitan Area, Minnesota, 1970-79: U.S. Geological Survey Open-File Report 84-577, 130 p.

____ 1986, Development of a water-use data system in Minnesota: U.S. Geological Survey Water-Resources Investigations Report 85-4306, 59 p.

Minnesota Department of Jobs and Training, 1986, Average covered employment and wages by economic region for calendar year 1985 by two digits of industry: St. Paul, 45 p.

Minnesota Land Management Information Center, 1984, State water use data system (SWUDS) system documentation: State Planning Agency, St. Paul, 42 p.

Office of Management and Budget, 1972, Standard industrial classification manual: Executive Office of the President-Office of Management and Budget, Statistical Policy Division, Washington, D.C., 649 p.

Snavely, D.S., 1986, Water-use data-collection programs and regional data base of the Great Lakes-St. Lawrence River Basin States and Provinces: U.S. Geological Survey Open-File Report 86-546, 204 p.

Trotta, L.C. [1988a] Sources of water-use data in Minnesota: U.S. Geological Survey Open-File Report 87-544, 2 p.

_____[1988b] Aggregation of Minnesota water-use data and transfer of data to the National Water-Use Data System-Procedures and programs: U.S. Geological Survey Open-File Report 87-40, 16 p.

Young, P. G., 1987, An analysis of water use in Minnesota: Department of Natural Resources, Division of Waters Report to the Legislative Commission on Minnesota Resources, St. Paul, 49 p.

DEVELOPMENT OF A DATA BASE TO ACCOMMODATE MANAGEMENT OF WATER-RESOURCES DATA WITHIN A GEOGRAPHIC INFORMATION SYSTEM (GIS)

--

By Douglas D. Nebert and Joel Frisch

--

ABSTRACT

The U.S. Geological Survey is working with the U.S. Army Corps of Engineers Engineer Topographic Laboratories' Terrain Analysis Center (TAC) to establish a water-resources data base for areas outside of the United States and to integrate the data base with geographic information systems (GIS) technology. TAC wants to automate production of its standard cartographic products and build an extensive attribute data base by use of an integrated spatial data-base design. Such a design would improve the relation between the previously distinct cartographic and tabular data so that production and update of overlays and related tabular data may be conducted more efficiently. Managing the data within a GIS would permit water-resources data-base users to define and generate products in a variety of map scales, formats, and media with minimal lead time, and would provide analysts the opportunity to query the coordinated data base to support spatial and tabular retrievals.

INTRODUCTION

The U.S. Geological Survey is working with the Terrain Analysis Center (TAC) of the U.S. Army Corps of Engineers to design and implement a relational data model for the Department of Defense Water Resources Data Base (WRDB). The automated spatial data base will be used by TAC to generate cartographic and tabular products for use by all military forces. The U.S. Geological Survey's efforts since 1983 have included the preparation of a user-needs assessment, preliminary data-base design using the Oracle data-base management system, hydrologic analyses and data preparation, data-input form coding, retrieval testing, and product generation for test data associated with water resources and facilities in the Middle East.

The data base was originally intended to contain only text attributes for manually-produced map overlays keyed to Department of Defense 1:250,000 scale maps. These overlays portray the location, quantity, and quality of surface-water features (such as canals, streams, and reservoirs), ground-water features (such as wells, springs, and qanats), areas of ground-water development potential, and water- related facilities such as pumping stations, desalination plants, and pipelines. These overlays are used by military planners, engineers, logisticians, and TAC teams to identify available or potential water resources in an area for treatment, storage, and distribution.

As a distinct and separate entity from the cartographic manuscript, the tabular data base in Oracle has a restricted ability to handle spatial queries and makes it difficult to keep both the cartographic manuscript and data base current. Modification to either the map manuscript or the tabular data base requires that each be manually updated in a timely fashion. Having both cartographic and tabular data on one medium would eliminate duplication of data-management responsibilities and reduce confusion in update and analysis. For this reason, a data base containing spatial- and tabular-data elements was determined by the Survey-TAC team to be the most effective solution to resolve coordination problems in production.

A user-requirements study was conducted by U.S. Geological Survey and TAC personnel to evaluate the need for, and the uses of, water- resources data by the Army and other Department of Defense users. As a result of that study, a report by Frisch and Stephens (1985) identified the need for a computerized water-resources data base to provide source data for thematic-overlay production and facilitate interactive retrievals of tabular data to support user queries. The report indicated that consideration be given to using computerized GIS technology to produce overlays and manage tabular data. It also indicated the need for, as the next step, the preparation of a prototype or pilot project to include development of a sample data base and analytical reports, specification of a functional hardware/software environment in which such work could be done routinely, and a test of the products' use to analysts, planners, and other users.

Pilot Study: Appraisal of a Geographic Information System in Overlay Automation

A pilot study was initiated by the U.S. Geological Survey late in 1986 to assess feasibility of TAC to implement an integrated spatial and tabular data base using GIS software and techniques. Three topical overlays (ground water, surface water, and water-resource facilities) to the Al Kuwayt, Kuwait 1:250,000-scale map were prepared using the ARC/INFO software package. High-resolution overlays were produced in an approximation of the official TAC format specifications. A workplan was prepared by the U.S. Geological Survey to define incorporation of the GIS data base into the TAC program over the next several years. Based on an evaluation of the pilot study and investigations by the U.S. Army Corps of Engineers Topographic Developments Laboratory, TAC elected to proceed with GIS technology and acquired a copy of ARC/INFO for use in the preparation of map overlays for hydrologic and terrain analysis.

Implementation Phase

The implementation phase in the WRDB development, conducted by the U.S. Geological Survey includes the following work elements that are scheduled for completion in late 1989:

(1) Install and test GIS software and provide on-site training in the use of the software for the data-base implementation activities;

(2) Modify the existing data structure with respect to the inclusion and management of spatial water-resources features and associated attributes;

(3) Design and implement input, update, and product-generation modules for use with the data base;

(4) Design and implement a tracking system of map-overlay status for the WRDB;

(5) Provide guidance in the conversion of existing spatial and tabular-data manuscripts to the new data-base format;

(6) Conduct a full-system test of the prototype; and

(7) Prepare a user manual to document all aspects of data management in the system.

Elements 1 through 3 have been completed. Only element 2 will be discussed in detail in this report.

Purpose and Scope

The purpose of this report is to document the development to-date of the WRDB data model and to emphasize the utility of spatial (or GIS) approach to data-base management over tabular systems now predominately used. Requirements of TAC will be reviewed, and the spatial data-base design techniques used to meet these requirements are described. An overview of the resulting data model and its operating environment in a worldwide context is presented. Special attention is given to the consideration of spatial features as primary elements of the overall data model to which associated attribute data are linked.

PRIMARY ELEMENTS OF A WATER-RESOURCES DATA BASE

The water-resources data bases used by most agencies share some common characteristics that are inherent to the collection and storage of water-resources data collected at discrete locations. Although many data bases do not explicitly manage their data in a GIS data structure, the organization of the water data commonly includes both the observations and some type of spatial reference for each sampling site, typically stored as point locations.

Traditional Features

Traditional efforts of water-resources data-bases design have treated water features as unique point features, because agencies tend to manage data collected at specific geographic locations. The creation of a single site file that contains non-repeating, basic site information and one or more measurement files that contain repeating records of observations is a common theme in water-resources data bases. Measurement or observation data can be associated to the site file and its characteristics by a common key -- commonly a site identifier -- in both files. This hypothetical file structure is illustrated in figure E26. In this traditional example, the location of the site is stored in the site file, and occurs only once per station. Once linked to the site file, measurement data can be associated with that point in space.

Maintaining water-resources features as unique and separate entities allows standard retrievals to be made based on the attributes in either the site file or the measurement file. Lacking is the capacity to identify measurement sites on a stream network or within an aquifer boundary without explicitly encoding site codes into the site file. Without this important element, the capacity to identify or associate sites that meet common spatial situations is limited to data in the site file. However, identifying all possible common spatial situations is a lengthy and time-consuming task when done manually. In addition, the inclusion of too many elements in the site file can degrade retrieval performance and can require excessive amounts of data storage. The use of GIS techniques to populate the site file -- either statistically or dynamically -- can save substantial time in encoding site characteristics.

Site file

site identifier	station type	latitude	longitude
4201	gage	032345	0171856

Measurement file

site identifier	parameter	units	value	date	time
4201	temperature	degrees Celsius	24.3	830713	1430
4201	pH		7.2	830713	1430
4201	TDS	mg/L	300	830714	0930
4300	temperature	degrees Celsius	9.7	870203	1145

Figure E26 --Relational data diagram showing site and measurement files related by a common key item, “site identifier.” Arrows illustrate the one-to-many relation between a site and its measurements. Parameter type is stored in the column “parameter” and units of measurement are stored under “units.” TDS = total dissolved solids.

Benefits of Geographic Information Systems to Water-Resources Data Management

The use of a GIS as a spatial data-management system allows maintenance of traditional data files and logical associations between water features. This principle was recognized in the conceptual design of the future U.S. Geological Survey's data base, the National Water Information System (NWIS), version 90.1 (Edwards and others, 1987). Traditional water-resources features can be considered as elements of an infinite number of possible spatial sets; any of these can potentially be used to satisfy a query. Assignment of additional geographic characteristics to spatial entities can be accomplished through the use of GIS concepts of overlay (integration), adjacency, and connectivity with one or more spatial-data themes. For example, by overlaying or integrating measurement points with country boundaries, these points can be automatically considered members of different country sets and thereby be assigned the identity of the country they are in. Rather than manually encoding each feature with the name of the country it lies within, the GIS can be used to assign points with country identities, as needed. There can be as many potential sets as there are polygon or surface data sets by which point data could be classified--elevation, ground-water units, slope, and soil association are a few examples.

Adjacency can be used to associate features with other nearby features, and to provide new opportunities for retrieval classification. Point-in-polygon overlay is useful to assign polygon values to a point, but is limited to the literal position of that point with respect to the areas. Measurement sites can be flagged with the identity of the nearest stream segment or the nearest upstream gage, and the attributes of those related features can be accessible to the user for analysis or retrieval. Another use for the proximity calculation would be to identify all facilities near a given pipeline.

The connectivity of the various conveyances to one another should be clearly made, where known, to enhance the query of features along surface-water or conveyance networks. The Cartesian coordinates of a gaging station indicates little about its position within the stream network. Attributes such as direction (one- or bi-directional), the immediate upstream and downstream segments, and pointers to related features (for example, facilities such as pumping stations or wells) should be maintained where they can be determined. Such a connectivity or network query would be to request all attributes of wells associated with a given pipeline. To satisfy the query, all elements or segments of the pipeline in question would first have to be identified before all wells associated with these segments could be identified. The resulting list of wells then could be used in further analysis or to generate tabular printouts.

Other benefits of integrating cartographic and tabular data in a GIS include the ability to generate map products at a variety of scales, map projections, and with different degrees of generalization on a variety of map media. The management of all cartographic data on-line improves the turn-around time to produce a standard product.

Integrated management of the water-resources and cartographic data facilitates the production of non-standard cartographic and tabular products to satisfy a user query. Water features can be selected on the basis of their attributes in the data base (for example, depth greater than 100 meters, water use classified as agricultural). Data from other spatial data bases also can be used with water data within the GIS for specialized hydrologic analysis.

EVALUATION OF EXISTING DATA ELEMENTS AS SPATIAL FEATURES

An early task in the implementation phase was to characterize the known WRDB features as legitimate spatial features within the ARC/INFO data model. At present, these features include points, lines, and areas, but future versions of the GIS software also probably will allow nodes to have attributes, and will allow all feature types to be kept within one collection of data files (known as a coverage in ARC/INFO).

Data-Management Issues

The next step was to define the many potential associations among the features that may be used in standard queries of the data base. Several management issues needed to be resolved regarding the distinction between points and areas at a given scale, how node attributes could be managed in the current version of the software, and how areas of numerous facilities were to be managed.

Given that the scale of the original cartographic product is 1:250,000, the distinction between points and areas may become blurred. The size of a plotted point symbol on the final overlay could be equal to as much as 0.8 miles on the ground. Where a single coordinate for the feature is known, the feature is stored and generated as a point. Where a feature is determined to have area, multiple coordinates are stored in the data base and generated as a polygon.

Point features on streams, such as gaging stations, that should be managed as attributed nodes are managed at present as a "link" coverage--a digital data set of line and point data. When node attributes are supported by the GIS software at a future date, the data model can be modified and the nodes assume point attributes.

Areas that contain numerous water features too dense to depict as separate features on the overlay are displayed with text annotation on final map products that states "numerous canals" or other feature type. Data are not eliminated from the data base due to crowding at a given scale because most water-feature location information is scale independent. This allows for design of thematic map products at a wide range of map scales. The GIS potentially could suppress the display of some features based on feature density.

Functional Organization of Water Resources Data Base Spatial Features

A functional delineation of the WRDB features is shown in table E9. The majority of the spatial features to be managed by the data base are points, although several line features also exist. By logically grouping features on the basis of their function and feature type, the ability to retrieve information across features is enhanced. This can be considered a type of functional and spatial normalization.

Comparison of potential associations between the features in table 1 indicate that most groups of water features can be associated with any other one, by one characteristic or another. Wells can be associated with pipelines or storage tanks, canals with other canals. A retrieval option desired by TAC is the ability to select all water features related to a water feature of interest.

Table E9.--Department of Defense Water Resources Data Base features organized by functional similarity with spatial feature types

Feature grouping	Feature class	Spatial feature type
Processing facility	Desalination facility	point
	Waste-treatment facility	point
	Other purificationpoint facility	point
Storage facility	Water tank	point
	Water tower	point
	Small reservoir	point
	Cistern	point
	Surface-water body	area
Other facilities	Pumping facility	point
	Miscellaneous source	point
Conveyances	Stream (river)	line
	Canal	line
	Qanat	line
	Pipeline	line
Ground-water features	Well	point
	Well field	area, point
	Spring	point
Other symbols	Dam	line symbol
	Graphic leaders	line symbol
	Surface-water access reach	line symbol

Definition of Data Structure

Once spatial similarities were determined, the existing attributes were reviewed for similarity. Not surprisingly, many similar water features had similar, if not identical, data elements associated with them. Where functional features and attributes were identical, they were stored in the same data files.

Every water-resources feature in the WRDB has a basic set of attributes (much like the site file shown in fig. E26) stored in the identification file. This file contains unique reference information for the feature, such as the feature class (water tower, pipeline, and so forth.), official and common names, and a series of identification numbers. Every feature in the WRDB has a unique number, called REC, that can be used as one of the keys to associate with various data files.

Geographic location, as maintained in the WRDB, is in a file separate from the ID file because of the potential of having more than one reported coordinate and positional accuracy for the given feature. For example, imagery analysis can determine a general location for a feature, whereas subsequent field checking can yield a different location. If more than one location is identified for a feature, the professional judgement of the analyst is used to decide the coordinates used to actually generate the feature. Multiple coordinate references for a feature are managed as attributes of the feature.

Other files exist to store information about the source documents used, update transactions, and quality-assurance testing information. A security level also is stored with each record of each file in the WRDB to assist in producing cartographic and tabular products for audiences with different security classifications.

Diagramming the Normalized Data Structure

A simplified view of data-file organization is shown in figure E27. All files shown contain the common item REC and, therefore, can be directly associated with one another. The ID file is shown at the center to signify the central or unique data elements it contains. This file relates one-to-one (1:1) or one-to-many (1:M) with the feature-attribute tables across the top. Abbreviations and symbology are given in the figure explanation.

Each feature class (for example, water tower, pipeline, or spring) is described by the unique association of a set of related data files. All features have at least two files in common: the identification file (ID) and the remarks file (REMARKS). Additional files are used to carry other attributes in normal form.

Examples of the basic relations required for two different water features are given in figure E28. The top figure, E28A, shows most of the files needed to define a pipeline in the data base. Because a pipeline is a linear feature, its component arc (or arcs) is referenced in an arc attribute table and stored as internal ARC coordinates. Because a pipeline can be composed of several arcs, a 1:M relation is diagrammed. Multiple text remarks are stored in the REMARKS file.

There is a 1:1 correspondence between the ID file and the pipeline characteristics file (PIPELINE), whereas a potential 1:M association exists between ID file and the MEASUREMENT and RELATED FEATURES files. The RELATED FEATURES file is used to build cross references between any two associated WRDB features. This file contains the record number of the current feature and all related features, such as pumping stations or wells, with one record per association.

A well is diagrammed in figure E28B. This well, as a feature not associated with any other feature, has no entry in the RELATED FEATURES file, but is associated with WELL data. As a point feature, it has an entry in the point attribute table (PAT). Standard associations can be made between the ID and the REMARKS files.

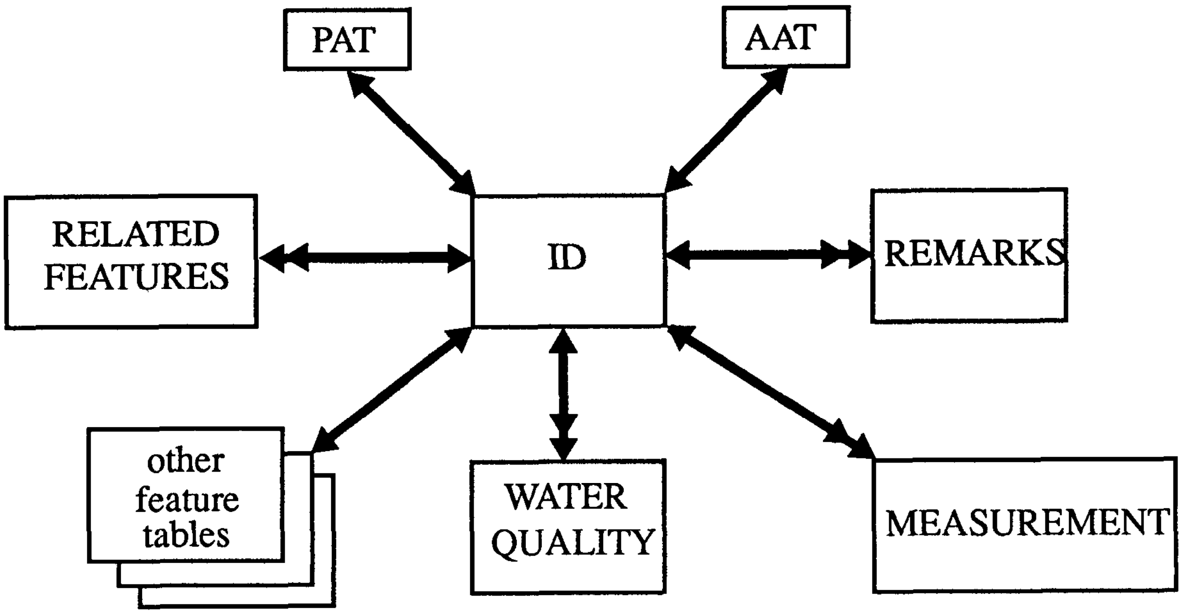
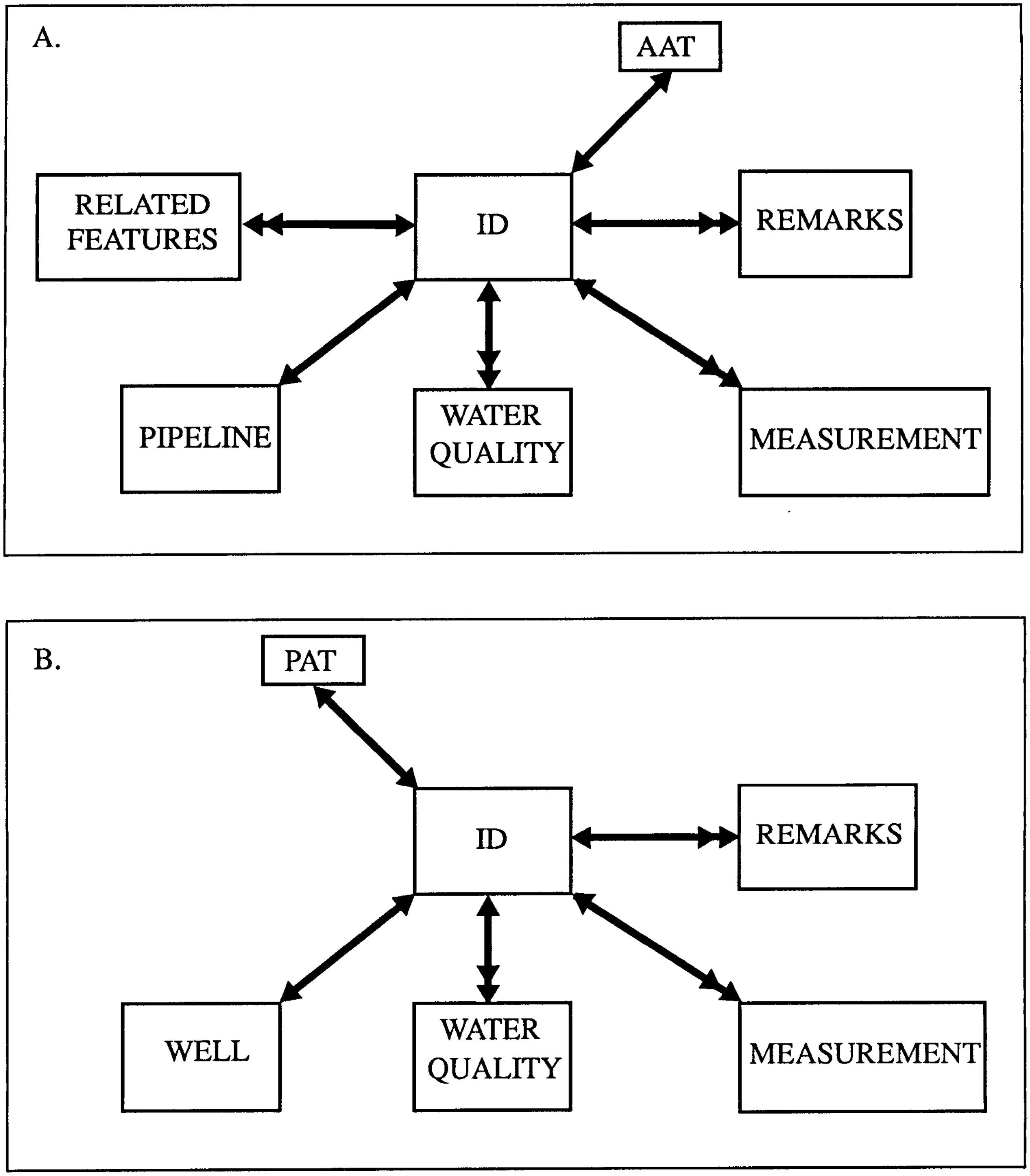


Figure E27 --Generalized diagram of data file organization and dependencies. Each file is represented by a box, arrows signify potential one-to-one and one-to-many relations. PAT = Point or Polygon Attribute Table, AAT = Arc Attribute Table. "Other feature tables" are tables for data unique to a given feature type such as a well. ID is the identity file that stores basic characteristics for all feature types.



Explanation

one-to-one association

one-to-many association

AAT is Arc Attribute Table
PAT is Point Attribute Table
ID is primary identification table

Figure E28 --Diagram of data file linkages for two water feature types: Figure E28A is for a pipeline feature; Figure E28B is for a well. Boxes represent data files; arrows represent file association by record.

GLOBAL DATA-BASE CONSIDERATIONS

This report has described the management of water-resources features on a single set of map sheet overlays. The management of a global data set requires the consideration of some additional organizational details. The first of these is the definition of the physical and logical organization of the data base on the computer system at TAC. As all WRDB data are compiled and stored with reference to 1:250,000- or 1:50,000-scale maps, the map boundaries define a logical data organization or map management (tiling) system for the WRDB.

To search the data base, users must know what maps lie within what countries in order to retrieve data in an area of interest. To prepare this cross reference, an unpublished coordinate file of map corners was acquired from Defense Mapping Agency. Coastline and political-boundary data were acquired from the Central Intelligence Agency's World Data Bank II at an approximate scale of 1:3,000,000 (Gorny, 1977). Both data types were converted into ARC/INFO coverages and the two were combined to produce an integrated global coverage that maintains a country code and a quadrangle number for each resulting map/country polygon. From the polygon attribute table, one can identify all maps that fall within each country and the percentage of each map within each country. Using the interactive GIS query functions, one can also identify all maps or countries within a circle, rectangle, or irregular polygon. This important basic capability is useful for browsing the data base. Each quadrangle-based workspace contains one full set of coverages and all associated data files.

There are several advantages to storing the data base in map-based tiles. By storing each map-based tile separately, continuity with the original map manuscript is preserved--including its original map projection. Entire workspaces can be copied to tape or to an analyst's work area without affecting the overall data base. Graphic selection of feature data can be done across tile boundaries, eliminating the need to append adjacent maps before analysis. Finally, the integrity of the standard map-based products is maintained. For analysis that requires several maps to be queried, the contents of adjacent tiles can be joined to produce new products.

Input, update, and plot generation operations are conducted by tile and are monitored via the map overlay status tracking system. The use of the Map Librarian package with the tracking system can facilitate the on-line management of hundreds of tiles planned for automation within the next few years.

CONCLUSIONS

The management of associated water-resources features is a logical task for spatial data-management functions available within state-of- the-art GIS software. The management of the extensive attribute data base can be done adequately within the relational data-base management systems (DBMS) used by GIS packages. As individual data files increase in size, the performance of these DBMS could be enhanced through the use of indexed data structures or through the use of intermediate key files that contain common retrieval items. Given the relatively small size of the tile-based data sets, these techniques have not been implemented.

The use of a coordinated spatial and tabular data base within a GIS processing environment is an ideal means of supporting the data requirements of TAC and its user community. The improvement of timeliness of data products, the versatility in producing them, and the added sophistication in spatial analysis are three areas where the use of GIS techniques should excel over manual methods of production.

As with the U.S. Geological Survey's NWIS 90.1 data-base design, TAC has recognized a need to manage its water-resources data in a single, integrated data model. Unlike the initial versions of NWIS design, TAC's WRDB redesign reflects the need to manage water features as primarily spatial entities, forming the basis of the data model. Because the objectives of the system are to analyze and manage the water-resources data in a specific geographic area, the GIS approach should improve access time, speed the production of map-based and tabular products, and enhance on-line analysis of water-resources data with respect to other digital spatial information (for example, geology, elevation, base thematic data from the Defense Mapping Agency).

SUMMARY

The U.S. Geological Survey and the U.S. Army Corps of Engineers are designing a water-resources data base that integrates extensive tabular data with geographic or mapped data. These data are used with a geographic information system (GIS) to produce standard cartographic products and to provide an environment for query and management of water-resources data.

A user-needs assessment was conducted to evaluate the need for and uses of water-resources data by Department of Defense users. GIS technology is the preferred means of managing the cartographic and tabular data defined in the user-requirements study. A pilot study was conducted on a 1:25,000-scale map of Kuwait to demonstrate the ability of a GIS to manage both spatial and tabular data in a coordinated manner.

Water-resources features defined in the user-requirements study were characterized as spatial features (points, lines, areas), and logical associations between features were made to define the appropriate data structure. Attribute data were normalized with respect to the water-resources features and other attributes to produce an optimal relational data base with most tables containing a common identification data item (REC). This common data item allows virtually any files to be related to support a tabular or graphic query in the GIS.

The GIS approach to water-resources data management will facilitate production of standard maps and tables and interactive analysis of the data with respect to other spacial data; this provides flexible user- views into the integrated data base.

REFERENCES CITED

Edwards, M.D., Putnam, A.L., and Hutchison, N.E., 1987, Conceptual design for the National Water Information System: U.S. Geological Survey Bulletin 1792, 22 p.

Frisch, Joel, and Stephens, Jerry C., 1985, User requirements analysis for the Worldwide Water Resources Data Base: U.S. Army Corps of Engineers, ETL Report ETL-TAC MISC 01-85, 50 p., plus appendixes.

Gorny, Alexander J., 1977, World data bank II, Volume 1 -- North America, and general user guide, Central Intelligence Agency Report CIA/DF-77/001a.

THE USE OF OPTICAL MEDIUM AS A MEANS FOR STORAGE OF IMAGE AND DIGITAL DATA

By Brenda L. Groskinsky and Richard A. Hollway

ABSTRACT

Storage of image and digital data, either for initial collection or after publication, is a requirement of all projects within the U.S. Geological Survey. Optical storage devices can be used as a means for archival storage and also as a means for temporary routine storage of image and digital data. Recent advances in optical storage devices indicate the need for applied research and evaluation of prototypes to assess the viability of optical medium for archival and temporary routine storage of image and digital data.

The major difference between the concepts of archival and routine backup of data occurs in the intent. An archive of data is a method to store historical record for periods greater than 10 years. The concept of backup, however, was devised as a precautionary measure in the event of data loss with information being stored for a period of less than 3 years. There are several physical alternatives for each concept. The objective of this research is to determine the advantages and disadvantages of optical or write-once-read-many drives as a method of image and digital data storage and retrieval when used as both an archival and a backup device.

The assessment of write-once-read-many drives has shown that the 32-bit Unix workstation allows integration of nearly all of the currently (1988) available write-once-read-many drives. At this time there is a limited amount of data base management system software that will store and retrieve both text and image data as well as interface directly to optical devices. Research concerning image scanners as input devices to the data management systems indicates Small Computer System Inteface (SCSI) scanners to be the most acceptable. The functions of a stand-alone optical storage and retrieval system would be isolated from the internal operations of the networked office environment commonly found throughout the U. S. Geological Survey. Recent advances in optical technology imply the adoption of a uniform write-once-read-many formatting standard, but possible interfacing difficulties remain between computing platforms. Because optical disk drive technology is still evolving, the longevity of a write-once-read-many disk could easily extend beyond the life-span of the optical drive.

Optical media have the ideal potential to provide a longer storage life for the archival of data than either paper or microfiche. Optical media also will work well as a means for temporary routine backup storage because of its high volume storage capacity. Therefore, future research of optical storage technology by the U.S. Geological Survey is needed.

CHAPTER F--ELECTRONIC PUBLISHING/MANUSCRIPT PREPARATION

AUTOMATING PROCEDURES FOR ANNUAL WATER DATA REPORT PREPARATION

By Mark L. Farmer and Jim E. Monical

U.S. Geological Survey

ABSTRACT

New procedures for automating many of the processes involved in the annual State water data reports of the U.S. Geological Survey have been developed by the Arkansas District of the U.S. Geological Survey. The automated process is based on the same steps as in manual compilation to establish a sense of continuity in preparation of the report. A directory structure was established to centralize the storage of all data files and software applications involved in the preparation of the data report. All software for the system were designed to be as modular and portable as possible.

Post-processing routines in Fortran 77 and Prime Command Procedure Language were developed to reformat data tables and retrieve from National Water Information System data bases as well as collating tables and text data. Macros have been developed using a full screen text editor for merging and sorting heading and table data.

INTRODUCTION

One of the primary functions of the U.S. Geological Survey is to collect data on the quantity and quality of surface and ground water and publish these data accurately and in a timely manner. Sophisticated methods are continually being developed for data collection and for recording and storing data with the use of high-precision recorders and high-speed computers. Although efficient hydrologic data collection and recording methods are essential, the need for efficient publication of these data also exists. The reporting of the data can be enhanced by automating some of the manual methods currently in use. The automation process incorporates the use of high-speed computers, retrieval and word-processing software, and laser printing technology.

District offices of the Geological Survey currently store most hydrologic data in the data bases of the National Water Information System (NWIS) of the U.S. Geological Survey. These data bases include the Automated Data Processing System (ADAPS), Ground Water Site Inventory (GWSI), Quality of Water (QW), and the New Site Specific Water Use Data System (NEWSWUDS). These data bases are supported through a network of Prime minicomputers, which uses Fortran 77 as the principal programming language for storage and retrieval applications.

Purpose and Scope

The purpose of this paper is to discuss the report preparation procedures developed in the Arkansas District of the Geological Survey, focusing on the preparation of the annual water data report. Also discussed are post-processing routines, written in Fortran 77, Prime's Command Procedure Language (CPL), and EMACS, Prime's full-screen text editor. These routines are used to reformat data tables retrieved from the NWIS data bases as well as collating table and text data for use in the annual water data report.

SOFTWARE DEVELOPMENT OBJECTIVES

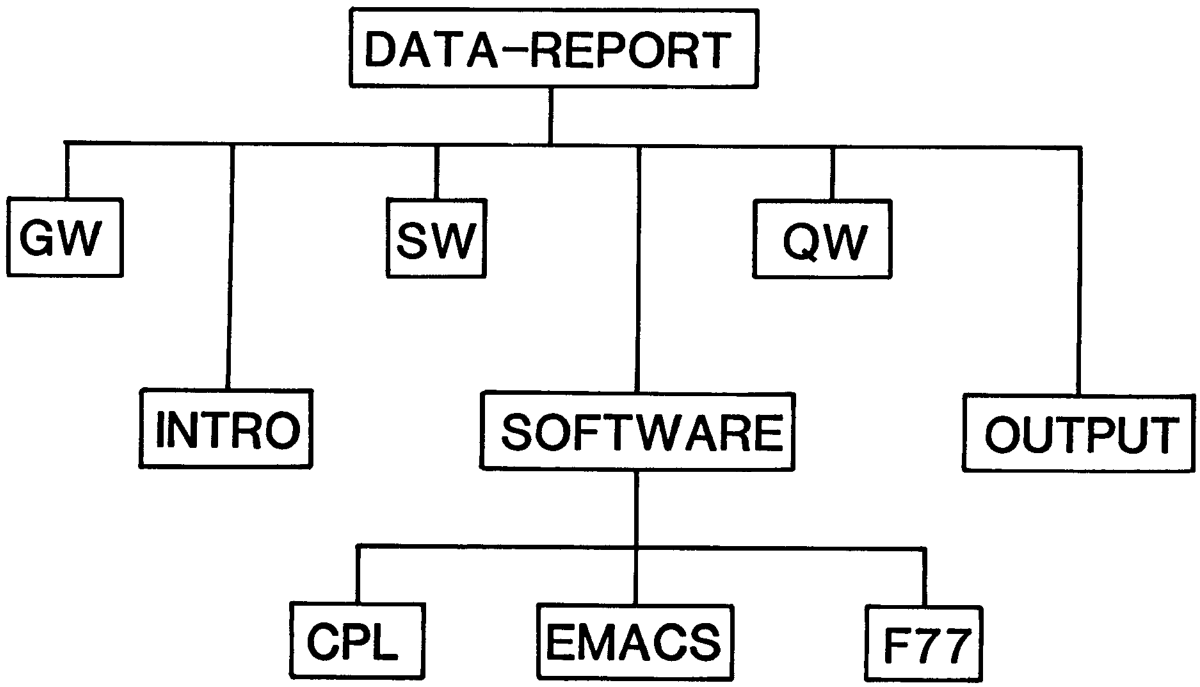
The steps used in the manual compilation methods of report preparation were used as a guide for the development of the automated system. Designing the system in this manner would allow a smooth transition for personnel adapting to the automated system, provide a guideline for software development, and would use experienced personnel who have worked with and developed the manual compilation system.

The first of two objectives, therefore, was to centralize the storage of all data files and software applications involved in the preparation of the data report into a top-level, user-file directory (UFD) on the Prime computer named DATA_REPORT. This UFD would contain six subdirectories that would group data and software files by subject (fig. F1). The SW, QW, and GW subdirectories contain all manuscript and tabular data for the surface-water, quality-of-water, and ground-water disciplines, respectively. The SOFTWARE subdirectory contains all of the application programs for the automated system, whereas the INTRO subdirectory contains all of the introductory text material for the annual report. The OUTPUT subdirectory is a repository for the final report after all text, station description, and table data have been compiled and merged. The centralization of all data files and software into one user-file directory allows control over the access to the data; this prevents accidental corruption or deletion of important files containing data or software.

The second objective was to ensure that all software developed for the automated system would be as modular and portable as possible. All of the software and file structures used in the automated system should be easily portable from one Primos partition to another and be transportable to other Prime systems. The only requirement for transferring software and data files to other systems is that the receiving systems contain the user-file directory DATA_REPORT, so that software references to that directory will successfully access the residing data files.

AUTOMATED WATER DATA REPORT PROCESS

The first step in the automated data report process is to update the introductory text using one of the Prime text editors or a word processing system software package. The textual data includes the introduction, summary, and formulas that are traditionally found at the beginning of the report. These files are placed in the INTRO subdirectory and, once established, require few changes from year to year.



EXPLANATION

GW - Ground water data tables.

SW - Surface water data tables.

QW - Quality of water data tables.

SOFTWARE - Contains all CPL, EMACS, and F77 programs.

OUTPUT - Contains camera ready water data report.

Figure F1.--Directory structure containing data and software used in the automated preparation procedures for the Annual Water Data Report.

Updating the station description text, which includes the downstream order number, station name and location, and period of record, is the second step. This text material complements the corresponding data tables for that particular station in the annual water data report. The subdirectories, SW, GW, and QW contain one large file of the station description text for that particular discipline with the exception of SW, which contains one station description file for partial record stations, and another for complete record stations. All of the station description files are arranged by downstream order number.

After the station description files have been updated, all data tables to be published in the report are retrieved. The ADAPS and QW systems, used to retrieve surface- and ground-water data from the NWIS data bases, are the primary systems used for this purpose. Retrievals are currently handled by authors of the data report on a station by station basis; however, the Arkansas District is currently developing programs to retrieve automatically all the data from both QW and ADAPS. Once the data tables are retrieved, the QW tables are then reformatted by a post-processing routine, which restructures the tables in a more suitable format to be used in the annual water data report. All tables retrieved then are placed in their appropriate discipline subdirectories (SW, QW, and GW).

The merging of all station description data with its corresponding table data then follows. This task is accomplished by first using an EMACS macro to search the manuscript files and isolate an individual station description. The table file is then searched to find the corresponding table data to merge with the station description text. During this process, extraneous lines (blank lines, duplicate data, and so forth) are deleted from the data tables, page headings are automatically generated, and page-division characters are inserted. The completed output files are written to the output subdirectory. The original data files are not disturbed so that if an error is discovered, a correction may be applied and the data can be processed again. The system was designed to be flexible enough to allow large table files to be subdivided into smaller files for processing.

The page numbering process is executed after the introductory material has been compiled and all station description headings are merged with their respective data tables.

The last two processes involve the creation of the table of contents and the index. Although the production of the table of contents is partially automated, the complete automation of these processes is still under development. An EMACS menu program is currently available to assist the preparer with the table of contents, but the index to the report needs to be written to a file using one of the available text editors on the Prime. Once the table of contents and the index file have been created, they are then appended to the finished output file and the file is printed using a laser printer to produce a camera-ready copy to be used for publication.

SOFTWARE

The following is a description of all of the programs used to automate the production of the annual data report. The languages used for the development of these programs are CPL, Prime EMACS Extension Language (PEEL), and Fortran 77.

CPL Programs

The automated procedure is driven by the CPL program, DATARPT.CPL, that displays the following menu options:

(1) Retrieve Data,
(2) Merge Quality Water Manuscripts and Tables,
(3) Merge Surface Water Manuscripts and Tables,
(4) Merge QW and SW files from (2) and (3),
(5) Merge Ground Water Manuscripts and Tables,
(6) Quality of Ground Water,
(7) Number pages in a file, and
(8) Exit and Return to Primos.

The first option in this menu executes the CPL program, RETRIEVALS.CPL, that performs data-table retrievals from either the water-quality data base, QWDATA, or the surface water data base, ADAPS. The other menu selections execute CPL programs that initialize and execute the appropriate EMACS macro for that selection. These CPL programs can be executed as either interactive or non-interactive processes. The programs perform edit-checking, such as testing for the existence of key input files, creating files that contain error messages. The user is allowed the option of selecting a page length or accepting a default. The page length is then passed on to the EMACS macro.

EMACS Programs

EMACS was chosen to perform all of the text processing functions because most of the work involved in the production of the annual data report consists of word processing tasks, such as manipulating blocks of text, and searching for unique character strings. The EMACS programs, called macros, are written in the programming language, PEEL.

The EMACS macros listed below perform the same text processing functions, merging manuscript data with corresponding data tables, for the class of data indicated:

DISCHARG.EM - Surface-water stations,
REGRPT.EM - "Complete" water-quality stations,
PARTRPT.EM - Parital-record water-quality stations, and
GWQW.EM - Ground-water water-quality stations.

Each program uses two input files, the first containing station description data and the second containing data tables for that discipline. Each of the input files are copied into a work area where they may be manipulated without altering the original file.

These programs isolate the station description data for a given site in a work buffer, extracting the site identification. By using the site identification and the station description, the programs construct the headings that will appear at the top of each completed page. The buffer containing all of the data tables then is searched for those tables associated with the current site identification. If those tables are present, they are copied into the work buffer. If no data tables are found for a site, a message is written to an error file and the program continues processing. The manuscript and table data then are merged and written to the output buffer, one page at a time. This process continues until all station description entries in the input file have been merged with their respective data tables.

The EMACS macro, PAGENUMBER.EM, numbers the pages in the output file. The program prompts the user for the beginning page number, left and right margins, and for which side of the page to start placing the page numbers. Given the margins and the number of digits in the page number, the program calculates the correct columnar position to place the page numbers on either side of the page. The program starts at the top of the file, searches for the page delimiter, and inserts the page number in the correct position on the next line. This process is continued until the end of the file is reached.

The macro TOC.EM creates the table of contents. The macro first reads the file containing all of the merged data and builds a separate file that contains the station name followed by the page number of that station. After the file is built, each line of the file is displayed, and the user is prompted to enter the number of spaces to indent that line from the left margin, and to enter the type of data that has been recorded for this station (discharge, chemical, biological, and so forth). The line then is reformatted with the station name indented the correct number of spaces, followed by the abbreviations for the types of data recorded, ending with the page number at the right side of the page.

Fortran 77 Programs

The program QTAB.F77 reformats the data tables output from the QWDATA system. This program deletes all occurrence of "no value" indicators (represented by '--') where the indicators are the only items present for a given time sequence in the data table (tables F1 and F2). The program realigns the columns of the output table for centering purposes, and also includes the 'TIME' column on every table within a given retrieval. QWDATA currently gives the 'TIME' column on only the first page of a station retrieval.

SUMMARY

New procedures for automating the preparation process of the annual state water data report have been developed by the Arkansas District of the U.S. Geological Survey. The automation process uses high-speed computers, retrieval and word processing software, and laser printing technology.

Table F1.--*Data table from QW data retrieval system before processing by QTAB reformatting program*

07032000 - MISSISSIPPI RIVER AT MEMPHIS, TENN.

WATER QUALITY DATA, WATER YEAR OCTOBER 1987 TO SEPTEMBER 1988

DATE	TIME	AGENCY COL-LECTING SAMPLE (CODE NUMBER) (00027)	AGENCY ANA-LYZING SAMPLE (CODE NUMBER) (00028)	STREAM-FLOW, INSTAN-TANEOUS (CFS) (00061)	SPE-CIFIC CON-DUCT-ANCE (US/CM) (00095)	PH (STAND-ARD UNITS) (00400)	TEMPER-ATURE WATER (DEG C) (00010)	TUR-BID-ITY (FTU) (00076)	OXYGEN, DIS-SOLVED (MG/L) (00300)
DEC									
28...	1230	80513	80020	748000	300	7.60	6.5	150	9.6
JAN									
26...	1330	80513	80020	762000	330	7.40	3.5	120	11.2
APR									
19...	1415	80513	80020	584000	419	8.00	15.0	48	9.7
JUN									
27...	1340	80513	80513	--	548	8.08	29.0	--	8.6
27...	1342	80513	80513	--	--	--	--	--	--
27...	1344	80513	80513	--	548	8.14	29.0	--	9.7
27...	1346	80513	80513	--	548	8.11	29.0	--	9.5
27...	1348	80513	80513	--	--	--	--	--	--
27...	1400	80513	80020	123000	548	8.10	30.0	17	8.2
JUL									
18...	1200	80513	80020	141000	492	7.80	30.0	37	7.5
AUG									
16...	1020	80513	80020	130000	538	8.30	31.0	18	8.1

07032000 - MISSISSIPPI RIVER AT MEMPHIS, TENN.

WATER QUALITY DATA, WATER YEAR OCTOBER 1987 TO SEPTEMBER 1988

DATE	SILICA, DIS-SOLVED (MG/L AS SIO2) (00955)	SOLIDS, RESIDUE AT 180 DEG. C DIS-SOLVED (MG/L) (70300)	SOLIDS, SUM OF CONSTI-TUENTS, DIS-SOLVED (MG/L) (70301)	SOLIDS, DIS-SOLVED (TONS PER AC-FT) (70303)	NITRO-GEN, NITRATE DIS-SOLVED (MG/L AS N) (00618)	NITRO-GEN, NITRITE DIS-SOLVED (MG/L AS N) (00613)	NITRO-GEN, NO2+NO3 DIS-SOLVED (MG/L AS N) (00631)	NITRO-GEN, AMMONIA TOTAL (MG/L AS N) (00610)	NITRO-GEN, AMMONIA DIS-SOLVED (MG/L AS N) (00608)	NITRO-GEN, ORGANIC TOTAL (MG/L AS N) (00605)
DEC										
28...	5.7	182	174	0.25	--	<0.010	1.20	0.140	0.150	0.36
JAN										
26...	5.9	195	199	0.27	1.68	0.020	1.70	0.130	0.110	0.67
APR										
19...	5.5	236	231	0.32	2.17	0.030	2.20	0.090	0.050	0.41
JUN										
27...	--	--	--	--	--	--	--	--	--	--
27...	--	--	--	--	--	--	--	--	--	--
27...	--	--	--	--	--	--	--	--	--	--
27...	--	--	--	--	--	--	--	--	--	--
27...	--	--	--	--	--	--	--	--	--	--
27...	2.9	303	310	0.41	--	<0.010	0.360	0.040	0.090	0.76
JUL										
18...	1.4	288	294	0.39	0.410	0.010	0.420	<0.010	<0.010	--
AUG										
16...	0.88	309	305	0.42	--	<0.010	0.300	0.030	0.030	0.67

Table F2.--*Data table from QW data retrieval system after processing by QTAB reformatting program*

07032000 - MISSISSIPPI RIVER AT MEMPHIS, TENN.

WATER QUALITY DATA, WATER YEAR OCTOBER 1987 TO SEPTEMBER 1988

DATE	TIME	AGENCY COL-LECTING SAMPLE (CODE NUMBER) (00027)	AGENCY ANA-LYZING SAMPLE (CODE NUMBER) (00028)	STREAM-FLOW, INSTAN-TANEOUS (CFS) (00061)	SPE-CIFIC CON-DUCT-ANCE (US/CM) (00095)	PH (STAND-ARD UNITS) (00400)	TEMPER-ATURE WATER (DEG C) (00010)	TUR-BID-ITY (FTU) (00076)	OXYGEN, DIS-SOLVED (MG/L) (00300)	SILICA, DIS-SOLVED (MG/L AS SIO2) (00955)
DEC										
28.	1230	80513	80020	748000	300	7.60	6.5	150	9.6	5.7
JAN										
26.	1330	80513	80020	762000	330	7.40	3.5	120	11.2	5.9
APR										
19.	1415	80513	80020	584000	419	8.00	15.0	48	9.7	5.5
JUN										
27.	1340	80513	80513	--	548	8.08	29.0	--	8.6	--
27.	1342	80513	80513	--	--	--	--	--	--	--
27.	1344	80513	80513	--	548	8.14	29.0	--	9.7	--
27.	1346	80513	80513	--	548	8.11	29.0	--	9.5	--
27.	1348	80513	80513	--	--	--	--	--	--	--
27.	1400	80513	80020	123000	548	8.10	30.0	17	8.2	2.9
JUL										
18.	1200	80513	80020	141000	492	7.80	30.0	37	7.5	1.4
AUG										
16.	1020	80513	80020	130000	538	8.30	31.0	18	8.1	0.88

07032000 - MISSISSIPPI RIVER AT MEMPHIS, TENN.

WATER QUALITY DATA, WATER YEAR OCTOBER 1987 TO SEPTEMBER 1988

DATE	TIME	SOLIDS, RESIDUE AT 180 DEG. C DIS-SOLVED (MG/L) (70300)	SOLIDS, SUM OF CONSTI-TUENTS, DIS-SOLVED (MG/L) (70301)	SOLIDS, DIS-SOLVED (TONS PER AC-FT) (70303)	NITRO-GEN, NITRATE DIS-SOLVED (MG/L AS N) (00618)	NITRO-GEN, NITRITE DIS-SOLVED (MG/L AS N) (00613)	NITRO-GEN, NO2+NO3 DIS-SOLVED (MG/L AS N) (00631)	NITRO-GEN, AMMONIA TOTAL (MG/L AS N) (00610)	NITRO-GEN, AMMONIA DIS-SOLVED (MG/L AS N) (00608)	NITRO-GEN, ORGANIC TOTAL (MG/L AS N) (00605)
DEC										
28.	1230	182	174	0.25	--	<0.010	1.20	0.140	0.150	0.36
JAN										
26.	1330	195	199	0.27	1.68	0.020	1.70	0.130	0.110	0.67
APR										
19.	1415	236	231	0.32	2.17	0.030	2.20	0.090	0.050	0.41
JUN										
27.	1400	303	310	0.41	--	<0.010	0.360	0.040	0.090	0.76
JUL										
18.	1200	288	294	0.39	0.410	0.010	0.420	<0.010	<0.010	--
AUG										
16.	1020	309	305	0.42	--	<0.010	0.300	0.030	0.030	0.67

The steps used in manual methods of report preparation were used as a guide for development of software in the automated system and also to establish continuity between the old and new report preparation process.

A directory structure was developed to centralize the storage of all data files and software applications. The directory structure contains six subdirectories that group data and software files by subject. The SW, QW, and GW subdirectories contain all text and tabular data from NWIS systems ADAPS, QW, and GWSI, respectively. The other subdirectories are: SOFTWARE, containing all application programs for the system; INTRO, introductory text material; and OUTPUT, the repository for the final report after text station descriptions and table data have been compiled and merged.

Post-processing routines have been developed in Fortran 77 and Prime Command Procedure Language to reformat data tables retrieved from NWIS data bases. These data bases include ADAPS, GWSI, QW, and NEWSWUDS. The post-processing routine will also collate table and station data suitable for use in the annual report. Macros also have been developed, using EMACS, to perform all text processing functions, merging manuscript with data tables, page numbering and creating tables of contents.

SELECTED REFERENCES

Buck, B., 1982, EMACS Reference Guide IDR 5026, Revision 18.3: Prime Computer, Inc.

Lamb, T.E., Porter, J.E., Lambert, B.F., and Plafcan, Maria, 1988, Water resources data for Arkansas--water year 1987: U.S. Geological Survey Water-Data Report AR-87-1, 575 p.

32-bit Workstations: The Trials, Tribulations, and Triumphs of Converting to an Open-System, Tools Environment

By David R. Boldt

ABSTRACT

The U.S. Geological Survey will use 32-bit supermicrocomputers to complement the existing minicomputer network and to provide for the increasing computer needs of the Geological Survey for the next several years. The Geological Survey office in Ithaca, NY, has been among the first to use a 32-bit workstation for project work and report processing. The computer has been installed, operated, and maintained by hydrologists with no previous experience in site administration. Once the new operating system had been learned, the 32-bit workstation proved to be an extremely powerful tool. The flexibility of the operating system combined with the efficient and productive windowing environment, has made the workstation the preferred computing environment for data manipulation and visualization, programming, modeling, and document preparation.

INTRODUCTION

Why Workstation, and Why Change Operating Systems?

Workstations are computers intermediate in power between microcomputers and minicomputers. They follow the microcomputer paradigm in that every user has a dedicated central processing unit (CPU), but workstations are typically linked together by high-speed networks that allow them to use disk resources and other peripherals that exist elsewhere on the network. A workstation typically has a high-resolution screen for display of graphics and support of a windowing environment, which is the most effective interface currently available for allowing a single user to access the multitasking capabilities of an operating system such as Unix. Workstations excel at providing computing capabilities for computer-aided design, scientific modeling, desktop report processing, and geographic information systems. Distributed CPU's ensure that intensive computational tasks will not affect other users, and permit the high level of graphics feedback that is an integral part of the processing environment.

Unix is a hardware-independent operating system based on the C programming language. This basis in C rather than in an assembly language has made Unix relatively easy to transport from one machine to another. The operating system is available on virtually all varieties and sizes of computers, from large mainframes to home microcomputers. As a result of this hardware independence, the Federal government as a whole, and the U.S. Geological Survey, in particular, have been looking to Unix as a way to avoid the constraints imposed by the proprietary operating systems of most computer manufacturers.

This paper relates the impressions of a geologist with a moderate computer programming background who has been involved with the procurement, installation, and use of a 32-bit workstation, and tries to convey the difficulties and benefits of using a workstation. The interactive graphics interface is not easily described, but must be experienced to be truly appreciated. Once mastered, the window-based interface greatly increases the productivity of computer operations.

At present (1988), workstations are of interest mainly to a rather specialized group who feel comfortable in front of a glowing phosphor screen, but if the U.S. Geological Survey follows its present course, workstations and Unix will affect nearly all Geological Survey employees.

The Windowing Environment

To a new user the biggest difference between a 32-bit workstation operating system and mainframe-style operating systems is the user interface. When mainframe computing evolved from batch-only mode to a time-sharing system in which multiple users could simultaneously access the computer, each user had the illusion of exclusive access to the computer. A multitasking, windowing environment takes this concept a step further and provides the illusion of having access to several computers at once. (In a networked environment this can be literally true.). In the most general sense, a window is a virtual terminal, and a single, oversized, high-resolution screen can have several active windows in which a user can manipulate text files, run programs, display graphics, and so forth. Windows can be moved, covered by other windows, uncovered, resized, or closed. Closing a window changes the appearance of that window into a postage-stamp sized icon that functions as a sort of bookmark. An icon takes up much less space on the screen than a window and can be opened at any time to resume work. Manipulation of a window has no affect on programs that may be running within it.

Windows are manipulated by a small, hand-held device called a mouse. The mouse moves over a 9 by 8 inch gridded pad, while a small light-sensor in the base of the mouse senses movement and transmits this to the computer. (Microcomputers commonly use a mechanical mouse that does not require a special pad.) The position of a pointer or small arrow on the screen corresponds to the position of the mouse. Pressing one of the buttons on the mouse. Pressing one of the three buttons on the mouse will invoke a response that depends on what the arrow is pointing to and which button is pressed.

The windowing environment enables the transfer of text from one area to another. For example, text from one window may be copied into another by highlighting it with the mouse, which copies it to a temporary buffer, and pasting it to the selected location by pressing a different button on the mouse. Complex commands can be generated on the command line by copying pieces of previous commands and file names that are visible on the screen. In addition, previous interactions within a window are saved as they scroll out of view and can be retrieved by scrolling backwards.

An example of the windowing environment in action would be a session using an interactive program debugger. The debugger, which when executed, creates a new window that displays several tiles, or subwindows, showing (1) the source code marked with symbols, (2) the last line executed and the position of existing breakpoints, (3) a menu selection of debugger options, (4) a listing of values of variables, and (5) a space for typing debugger commands. The output of a graphics program, for example, can be directed to a graphics window so that, as the program runs within the debugger, the effect of individual graphics calls can be seen on the drawing as it is created. If a program error is found, it can be corrected with an editor in another window. The program can be edited, recompiled, and re-executed by moving from window the window and either typing the appropriate commands or using the mouse. Workstations are generally acknowledged to provide the most productive programming environment available.

The convenience of multiple windows whose size and locations can be easily manipulated soon becomes addictive because it avoids the exasperation caused by being in a program and needing some piece of information that resides elsewhere in the computer system, but can be accessed only be terminating the program. Instead, one can bring up a new window to access the needed information. A window can be expanded to fill the entire screen, providing up to 140 columns by 52 rows of normal-sized text, or reduced as small as desired. Even the type style and size can be changed for a window, so that one could see more data, or take off one's glasses. Each person who has been using the system long enough to become familiar with it develops an individual style of window management; some people work with so many open and partially obscured windows that the screen appears similar to a paper-cluttered desk. The benefits of an environment that "belongs" to the person working within it may be difficult to quantify, but most certainly improve productivity.

Communications

Communication with other computers by terminal emulation through asynchronous communications ports, is relatively straightforward. A Unix terminal emulation utility allows one to define a port configuration as to baud rate, parity, and so forth, and label it with a single word. The terminal emulator supports the transfer of files without error checking, which has been fast and trouble free for filed of less than 300 lines. For longer files, however, we have noted buffer overflow at a transmission rate of 4,800 bits per second or higher and loss of data. To send longer files, or files for which error checking is important, the staff has used the communications package Kermit, which is available for many varieties of computers from large mainframes to home microcomputers.

The two 32-bit workstations in our office are linked by a local are network, ethernet, which enhances the speed and reliability of communication among systems. The U.S. Geological Survey Distributed Informations System II requires local area network capabilities for communications among workstations.

The Complexity of a Workstation Operating System

Every operating system has individual characteristics that result from the philosophy of those who designed it. Unix was developed by programmers and systems developers to facilitate their own work. Thus, Unix has commands to do many things that might otherwise require working with a conventional programming language. Unix has extensive support for manipulating text files, work processing, and computer programming.

Unix has a reputation for being somewhat user-hostile; one reason is that it is extremely terse. Many operating systems contain shortened forms or abbreviations for commands, but Unix contains only abbreviations, and most commands consist of two or three letters. The reason for this brevity is simply that short commands require less typing--no small consideration to a programmer who may use these commands hundreds of times each day. Another factor contributing to the difficulty of using Unix is that some actions such as listing a file may not have direct commands but result as a side-effect of a different command; for example, the "cat" command can be used to list the contents of a file to the screen but is actually the command to join files, shortened from "catenate." A second example is that the command to move a file and to rename a file are the same (mv); one being a move between directories, and the other a "move" within the same directory. Still another cause of unease for new Unix users is the sheer size of the operating system; it contains many capabilities that add power and flexibility that can understandably intimidate the beginner.

Most occasional users on the Prime computer use no more than a half-dozen Prime operating system (Primos) commands. This minimal interaction with the operating system should ease the transition between operating systems for most people. In fact, the windowing environment can be tailored to guide those uncomfortable with computers through any desired process by means of pop-up menus.

Most of the effort in adapting to a 32-bit workstation will be learning the new text editors and applications programs. Those who have an aptitude for programming will likely find the Unix environment extremely flexible and supportive of their efforts. The 32-bit workstation in the Ithaca, NY, office is the preferred platform for writing programs and for the type of computer activity required to support ground-water modeling. The people who will be most adversely affected by the change in operating systems will likely be those of medium computer skills, that is, those who need to do more than run a few existing programs but who have not generally been writing their own programs. The complexity of the operating system has been a problem in the business world as well and has slowed the acceptance of Unix in the marketplace. As a result, workstation vendors have a strong interest in making Unix more accessible.

Capabilities of a Workstation Operating System

One of the most powerful features of Unix is Input/Output (IO) redirection. For example, program output that would normally be written to the screen can be directed to a file, and conversely, a program that would normally read from the terminal can be directed to read from a file. Fortran file unit 5 is reserved for standard input, whereas unit 6 is reserved for standard output. All programs adhering to Unix system conventions read from standard input and write to standard output to facilitate IO redirection. The idea of interactively routing IO is most powerfully demonstrated with "pipelines," whereby the standard output from one program is connected to the standard input of another so that results from one program can be used as data by another without the use of intermediate files. Any number of programs can be linked in this manner, each functioning as a filter on the data produced by the previous program. Below are some examples of how IO redirection is accomplished.

ls	--	list files in current directory
ls > *suds*	--	place listing into a file called "suds"
sort < *data* > *report*	--	sort the file "data" and place the results into a file called "report"
ls | sort -r	--	list files in current directory in reverse alphabetical order

The operators for IO redirection are:

<	--	use data from the file whose name follows the "<"
>	--	send output to the file whose name follows the ">"
|	--	connect the output from the program preceding the "|" to the input of the program following the "|"

To give some idea of the tools available in Unix, a few Unix commands are described below.

cut *cols filename*	list specified columns from a file
grep *text filename*	list all lines of a file containing "text"
head -n *filename*	list the first n lines of a file
paste *cols filename*	merge two files side by side
sort *filename*	sort the contents of a file
spell *filename*	invoke a spelling checker on a file
su *username*	switch to another userid without logging out
tail -n *filename*	list the last n lines of a file
tr *pattern filename*	apply global edit changes to a file
wc *filename*	count letters, words, and lines

This is just a sample of the variety of commands available. In addition, IO redirection can be used with any of these commands, wildcards may be used to specify groups of files, and pattern-matching wildcards are available as well.

As one gains experience with Unix, one's perception of the interaction between programs and files changes subtly. In most operating systems, files can be conceptualized as being stationary receptacles of information, and programs can be imagined to search for these files and perform actions on them. In the Unix environment, a more useful concept is that the programs are stationary and that the files themselves are in motion, carrying information between programs.

Another powerful feature of Unix is "last command editing," where all previously executed commands are stored in a a queue and may be listed, retrieved, edited, and reissued with a minimum of keystrokes. For example, to rename a file a fumble-fingered (or CTR hypnotized) programmer may have typed:

```
mc sf.data stream-flow-measurements
```

Having mistyped the "mv" command, the mistake can be corrected and the command reissued by typing:

```
^mc^mv
```

which results in the execution of the command:

```
mv sf.data stream-flow-measurements
```

One can also retrieve any of the words used in a previous command. For example, $! refers to the last word of the last command. If the following were typed immediately after the previous command:

```
cat $!
```

The command:

```
cat stream-flow-measurements
```

would be executed, and the file would be listed to the screen.

APPLICATION OF 32-BIT WORKSTATIONS

The Ithaca, NY office of the U.S. Geological Survey has used a 32-bit workstation for several project-related tasks during the year in which it has been available. These tasks are described below.

Ground-Water Modeling

A three-dimensional ground-water flow model by McDonald and Harbaugh (1988) has been installed and used to prepare a cross-sectional simulation of an aquifer response to an aquifer test. It is currently being used to prepare a series of cross-sectional models to estimate the hydraulic conductivity of a riverbed and also in the expansion of a two-dimensional ground-water flow model of a contaminated industrial site in western New York to a full three-dimensional ground-water flow model.

Data Base

A special-purpose data-base program has been written in the C programming language that is keyed to the bedrock stratigraphy in the area of Niagara Falls, in New York and Ontario, and includes stratigraphic horizons, elevations, and formation names. The data base is used to maintain information on formation pressures and geochemical data obtained from multilevel piezometer systems (12 to 21 zones per hole) at 17 locations. The intended use of this keyed file system is to produce maps of individual zones between the 17 well locations and to support the development of a three-dimensional ground-water-flow model.

Programming

Several Fortran programs have been ported to the 32-bit workstation from minicomputers. Most required minor alterations to run on the workstation, depending on the degree to which the programs adhered to standard American National Standards Institute (ANSI) Fortran 77. In many ways the software migration between the two systems has served to highlight exactly which features constitute standard Fortran 77.

The productive programming environment makes the development of programs much easier and faster on 32-bit workstations than on conventional minicomputers. A Unix facility named "make" uses information about the interdependency of subroutines so that, after a program is edited, only those subroutines that are directly affected by the change are recompiled. Another Unix utility named "SCCS" keeps track of all modifications made to a program and thus provides an audit trail so that one can retrieve previous versions or see who has edited a program.

Reports Processing

The 32-bit workstation with reports processing software is replacing outmoded, dedicated word-processing machines. The ability to create and edit on screen special characters; such as Greek symbols, italics, superscripts and subscripts; to create mathematical equations; and to mix text and graphics greatly facilitates the preparation of technical documents. The high quality of the financed product, produced by a laser printer, enhances the appearance of memorandums and correspondence.

The 1988, U.S. Geological Survey annual data report for Ithaca, NY, was prepared using the new hardware-software combination for the first time and was completed on schedule. Most of the time spent was on transferring and reformatting files that were stored on the dedicated word-processor.

Graphics

Graphics software is being used to generate graphs from data stored in a custom-built data base program. The graphics package is a C subroutine library that supports two- and three-dimensional graphics, but lacks facility in font handling and labeling. Interactive graphics packages are available for 32-bit workstations and the office staff has strong interest in the capabilities of interactive graphics software for preparation of charts and diagrams for presentations and reports.

SUMMARY

Unix workstations provide both a promise and a challenge. The promise is that each of us will have the capability of a dedicated computer and at our desk with the proper tools and interface to do our work much more efficiently than at present. The challenge is to manage the complexity of the new power and capability so that we retain what we have already gained while benefitting from new technology.

EVALUATION OF THREE ELECTRONIC REPORT PROCESSING SYSTEMS FOR PREPARING HYDROLOGIC REPORTS OF THE U.S. GEOLOGICAL SURVEY, WATER RESOURCES DIVISION

STILTNER, Gloria J., USGS, WRD, 12201 Sunrise Valley Drive, Reston, VA 22092

In 1987, the Water Resources Division of the U.S. Geological Survey undertook three pilot projects to evaluate electronic report processing systems as a means to improve the quality and timeliness of reports pertaining to water-resources investigations.

The three projects selected for study included the use of the following configuration of software and hardware: Ventura Publisher software on an IBM model AT personal computer, PageMaker software on a Macintosh computer, and FrameMaker software on a Sun Microsystems workstation. The following assessment criteria were to be addressed in the pilot studies: The combined use of text, tables, and graphics; analysis of time; ease of learning; compatibility with the existing minicomputer system; and technical limitations. It was considered essential that the camera-ready copy produced be in a format suitable for publication. Visual improvement alone was not a consideration.

This report consolidates and summarizes the findings of the electronic report processing pilot projects. Text and table files originating on the existing minicomputer system were successfully transferred to the electronic report processing systems in American Standard Code for Information Interchange (ASCII) format. Graphics prepared using a proprietary graphics software package were transferred to all the electronic report processing software through the use of Computer Graphic Metafiles. Graphics from other sources were entered into the systems by scanning paper images. Comparative analysis of time needed to process text and tables by the electronic report processing systems and by conventional methods indicated that, although more time is invested in creating the original page composition for an electronically processed report, substantial time is saved in producing subsequent reports because the format can be stored and re-used by electronic means as a template. Because of the more compact page layouts, costs of printing the reports were 15 to 25 percent less than costs of printing the reports prepared by conventional methods.

Because the largest report workload in the offices conducting water-resources investigation is preparation of Water-Resources Investigations Reports, Open-File Reports, and annual State Data Reports, the pilot studies only involved these kinds of products.

EVALUATION OF A DESKTOP REPORTS PROCESSING SYSTEM FOR PRODUCING EARTH-SCIENCE TECHNICAL REPORTS

By Richard A. Hollway and Denise A. Wiltshire

ABSTRACT

An evaluation of effectiveness of desktop reports processing systems for widespread application in the U.S. Geological Survey was conducted from November 1987 through June 1988 in association with two other studies. For this study, the hardware platform of the desktop reports processing system consisted of a 32-bit, multiuser microcomputer manufactured by Sun Microsystems, Inc.[1], (model 3/160). FrameMaker (version 1.11) software, manufactured by Frame Technology, Inc., was selected for the evaluation because it is an integrated text processing and page composition package that operates on a Sun workstation, and it is reasonably priced.

The evaluation process entailed producing a 92-page technical report with 13 tables and 41 figures using integrated desktop reports processing techniques and comparing procedures to those for preparing the same report using manual and conventional word-processing methods. The desktop reports processing system was evaluated by testing procedures of text processing, graphics, page composition, document management, and the user interface. In comparison to presently used word processing software, FrameMaker software provides excellent text processing and good-to-excellent page composition capabilities. Strengths of FrameMaker software include versatile document structuring, report format templates, and use of the SunView windowing interface. FrameMaker was consistently fast and accurate for global changes of text and page composition. The software offers easy to use pull-down and pop-up menus for selecting typefaces, column layout, and headers and footers. Among FrameMaker's best features is the more complete and faster Spelling Checker utility.

The hardware platform enhanced the performance of the software throughout the text processing procedures used during the evaluation . The 32-bit, virtual memory operating system of the Sun workstation offers broad functionality for an integrated word processing and page composition software package such as FrameMaker. Because FrameMaker operates within the SunView windowing environment, multiple documents may be viewed on the workstation screen and are easily copied into FrameMaker windows. In addition, facing pages of a document may also be viewed on the screen simultaneously, thus facilitating page layout. The virtual memory addressing system of the Sun workstation allows nearly limitless document sizes, and the power of the processing unit is sufficient to ensure rapid loads, saves, and global changes of text and page composition in reports exceeding 100 pages.

Comparisons of desktop reports processing methods to conventional document processing methods used by the U.S. Geological Survey indicate additional advantages of a fully integrated software package such as FrameMaker. Complex document and graphics transfers from a variety of computers and manual cut and paste procedures are eliminated using FrameMaker. The use of proportional typefaces, widow and orphan control, and condensed tables using tab construction techniques resulted in reducing the total number of pages of the final version of a document when compared to conventional document processing methods. As a result, a 20 percent decrease in printing costs may be expected using desktop reports processing software. Although additional time is required for preparing structured templates for page composition during the initial typing stage, the ability to store templates allows rapid production of subsequent reports with a similar format. Hence, substantial time can be saved if a publications office produces many reports requiring the same format specifications. Although improving the visual appearance of report formats was not an objective of the study, the style of the final publication was enhanced and printing costs were reduced when compared to conventional publishing methods.

USE OF AN ELECTRONIC PAGE-COMPOSITION SYSTEM TO PREPARE CAMERA-READY COPY OF SCIENTIFIC REPORTS

By L.H. Geiger[1], P.R. Mixson[1], and S.D. Flagg[1]

ABSTRACT

The U.S. Geological Survey publishes each year several thousand reports, abstracts, and articles that require time-consuming collaboration among authors, cartographers, illustrators, typographers, and editors. In 1989, camera-ready copy for approximately 550 of these reports was prepared using various word-processing software packages and desktop printers.

Three pilot projects were conducted to assess the feasibility of electronic page-composition systems to prepare camera-ready copy. Electronic page-composition was expected to result in low-cost and high-quality documents. Each project tested a different configuration of hardware and software. The projects were conducted in the Florida District, Ohio District, and Oregon offices of the Water Resources Division, U.S. Geological Survey.

In the Florida project, the configuration consisted of a personal computer with a full-page monitor, laser printer, word-processing software, and page-composition software. The scope of the Florida project included the preparation of camera-ready copy of several Water-Resources Investigations Reports, Open-File Reports, an annual State data report, text for map reports, and materials for slide presentations.

In assessing the electronic page-composition system in the Florida project, the following improvements were noted over previous systems:

- *Reusable style sheets reduce the amount of time spent on subsequent reports.*
- *Automatic preparation of table of contents, indexes, and page numbering reduces the amount of time spent on these parts of the report.*
- *Proportional fonts reduce the number of pages per report; this can lower printing costs. The greatest reduction of pages occurs in those reports having a high ratio of text to tables or illustrations.*
- *Importing graphics reduces the amount of time spent to integrate graphics with text and tables when preparing camera-ready copy.*

[1]U.S. Geological Survey, Tallahassee, Fla.

EVALUATION OF A USER-FRIENDLY ELECTRONIC REPORT PROCESSING SYSTEM FOR PREPARATION OF SELECTED REPORTS

By Michael Eberle
U.S. Geological Survey, Water Resources Division
Columbus, Ohio

ABSTRACT

An evaluation of a user-friendly electronic report processing (ERP) system showed that the system is easy to install and learn, readily transfers text data files to and from a minicomputer located in the same office, and adequately performs most of the text-processing and page-layout functions needed to prepare camera-ready copy for selected reports of the U.S. Geological Survey, Water Resources Division. The preparation of camera-ready copy of text and tables for two sample reports by use of the ERP system, however, took about four times longer than preparation of the same text and tables by use of a conventional word-processing system.

The evaluation, which was conducted by the Ohio (Columbus) office of the U.S. Geological Survey, was one of three evaluations that addressed basic questions of cost effectiveness and technical feasibility of introducing ERP systems for applications where use of conventional word-processing techniques has been the norm. The ERP system tested in the Ohio office consisted of Macintosh SE computer equipped with a 20-megabyte internal hard disk; an Apple LaserWriter printer; PageMaker page-layout software (version 2.0a; version 3.0 was not available during most of the evaluation); and ancillary terminal-emulation, word-processing, and graphics software. The conventional word-processing system consisted of an IBM-compatible microcomputer (also with a 20-megabyte internal hard disk), a 55-character-per-second strike-on printer, and WordMARC word-processing software.

An important assumption in the evaluation was that the ERP system should readily transfer text files to and from the network of minicomputers already in place. To test for transferability, all text and table files for the sample reports were transferred from the minicomputer colocated in the Ohio office to the ERP system in American Standard Code for Information Interchange format. Selected files also were transferred from the ERP system to the minicomputer.

For both sample reports prepared with the ERP system, text and table processing time was about equally divided between reformatting of American Standard Code for Information Interchange format files for transfer to PageMaker software and on-screen page layout; however, the shorter of the sample reports (which was entirely in a two-column format) took twice as long per page to complete as the longer report (which contained about 50 pages of data tables in a fixed-space font in single columns).

Evaluation of graphics was restricted to how well files in MacPaint format, PICT format, and Tag Image File Format could be transferred to the PageMaker software. Time and cost comparisons of ERP and conventional illustration-preparation techniques were not feasible. In general, graphics generated with the Macintosh-based software in all of the above-mentioned formats were transferred satisfactorily. Attempts to use images saved from the Macintosh screen in MacPaint and PICT formats, however, resulted in graphics characterized by garbled text and insufficient resolution for publication.

INDEX

☆U.S. GOVERNMENT PRINTING OFFICE 1991-550-938