

# DATA MANAGEMENT PROTOCOL



DENALI NATIONAL PARK AND PRESERVE

National Park Service  
U.S. Department of the Interior

## TABLE OF CONTENTS

PLANNING AND DESIGN .....	1
Year 2000.....	2
File Names .....	3
Data Exchange Formats .....	4
COLLECTION AND DATA ENTRY.....	4
Data Collected on Field Forms.....	5
Electronically Collected Data .....	6
Data from External Researchers.....	6
Computer-recorded data .....	6
Hand-recorded data.....	7
Instrumentation.....	7
DOCUMENTATION.....	7
Software .....	7
Data Dictionary .....	8
Metadata.....	8
QUALITY ASSURANCE.....	9
Editing Procedures.....	9
ACCESS .....	9
SECURITY.....	9
Field Form Security .....	9
Power 10	
Data Security .....	10
Backing Up Data Files .....	11
Tracking Back-Up Files .....	12
Virus Protection.....	12
FINAL DATA AND ARCHIVES .....	13
Final Data.....	13
Sensitive Data.....	13
Freedom of Information Act .....	13
Archiving .....	14
APPENDICES .....	15
Appendix A - Sample Spatial Metadata Entry .....	16
Appendix B - Sample Non-Spatial metadata form.....	20
Appendix C - Backup Diskette Tracking Form.....	21
Appendix D - Data Entry Procedures .....	22

Appendix E - Data Verification Procedures ..... 25  
Appendix F - Data Validation Strategies ..... 27  
Appendix G - Guidelines for Data File Editing ..... 31  
Appendix H - Guidelines for Disseminating Data ..... 38



DENALI NATIONAL PARK AND PRESERVE  
DIVISION OF RESEARCH AND RESOURCE PRESERVATION

DATA MANAGEMENT PROTOCOL

A system for data management is an essential part of any project where pieces of information must be organized into a reliable, useable form and stored in such a way as to be available for future use. A standardized, systematic approach is necessary to minimize error in the data and assure its longevity. The proper management of data can be divided into a number of phases, Planning and Design, Collection and Data Entry, Quality Assurance, Documentation, Access, Security, and Archiving.

Natural resource data is collected by a number of different programs and activities engaged in the management of Denali National Park and Preserve. The Long-Term Ecological Monitoring Program (LTEM), wildlife, vegetation, fire, and hydrological management programs are collecting resource information for the park overall. Outside researchers from other agencies, universities, and private organizations working under permit or contract are other contributors of resource data.

The purpose of this protocol is to provide standards for the management of digital data pertaining to the natural resources at Denali to ensure the protection, reliability and availability of that data. Other protocols will be developed to address maps, specimens, photographs, etc. As with the data it addresses, this is a dynamic document, subject to periodic updating as improved procedures are developed.

### **PLANNING AND DESIGN**

In order to simplify tasks later on, sufficient thought and planning must be expended in this phase of the effort. The needs of the end user must be taken into account. This will ensure that the right kind of data is collected. Data is to be recorded in a way that will provide for ease of entry in the field as well as transfer to the database back at the office. To minimize error all data must be gathered in a format that can be directly entered into a computer with no transcribing.

Standardized data forms effectively identify what pieces of data are to be gathered and can display data for efficient computer entry. Forms should be standardized across disciplines. Advance thought must be given to the order in which the data are gathered, units of measurement, and how much space is needed to write the data. Forms should be clearly titled as to the project, location, principle investigator, and the sort of data it contains. They should include space for the date, data observers and recorders, page number out of the total number of pages, and any other pertinent information. This will insure that a misplaced data sheet can be properly identified. Since first attempts at a field form will almost certainly need some revision, the revision date, file name, and where the form is stored (if produced from a computer) should also be included on the form.

Database file structure should be clean and simple. "Normalization" is a database term that describes the process of reducing redundancy and improving efficiency and speed by breaking existing database files into smaller, more easily managed files which link through key fields back and forth to other files which define or depend upon them. For example, a non-normalized file structure for a fish study would have each single record include fields for the date, full stream name, site, UTM coordinates, park district, and 20 species fields -- one for each fish likely to be encountered. This is redundant, and a better structure would be to have a single stream file containing specific site information (such as a code name, UTMs, elevation, etc.), and another with only four fields: site code, date, fish code and number collected. Here, if an error in a UTM coordinate is discovered it only needs to be changed in one place, rather than in every record for that site. Through proper planning a standard, normalized structure could encompass all park natural resources--linking aquatic, vegetation, geology, weather, and human impact data through a common system of smaller, well-defined and easily maintained files.

Databases do not necessarily need to be broken up into separate files based on time periods (for example, one database for each month or year); keeping all data in one database is usually more convenient. Subsets of data (for example, the current year or a particular research site) can easily be obtained using queries. When a dataset is merged into one covering a number of years, however, it is important to keep a field in the data file for "current year". For studies with extremely large quantities of data, however, the "current year" design is more efficient. A new version (empty table or database file) is started at the beginning of each research year. At the end of the year, the final (checked and approved) records can either be saved as a separate file or else appended to a master database of all records for all previous years. This approach has several advantages:

- 1) "Previous" and "final" data remain separated from "current, in progress" data.
- 2) Data file backup operations are simplified ("previous year" data need only be backed up once, at the close of each research year).
- 3) The numbers of data records in each file remain at a reasonable number.

### Year 2000

It is most likely that new software will address the need to ensure that date fields can handle the issue of the transition to the new millenium. However, this must be confirmed when using data management software and new software acquisitions must verify that the product addresses the year 2000 issue. As added insurance, all date fields must be formatted so those entries use four-digit year entry. Legacy datasets must be examined to ensure that all date fields comply with a four-digit year entry and be corrected as needed.

### File Names

Until recently, DOS-based software was constrained by the need to follow an 8.3 naming convention (eight characters followed by a three character extension). Some software packages such as WordPerfect allowed for more descriptive information as part of a document summary. The Windows 95/NT, as well as Macintosh and Unix, operating systems permit longer file names. For the time being, however, in order to maintain backward and cross-platform compatibility, file names will be restricted to the use of eight characters.

File names should indicate the contents of a file. When employing a current-year system, a date should always be included as part of the file name for the current year records. For example, RCCONE93.DBF identifies a data file containing information about spruce cones in the Rock Creek watershed in 1993. A "\_M", "MAST", or similar designation is appropriate for a master file (for example, RCCONE\_M.DBF).

Files containing data downloaded from data loggers on a frequent basis should include the year and "DAT" in their names. For example, PF93193.DAT identifies the file containing permafrost site data starting on 05/10/93 (Julian date 193).

Files containing the same data in different formats should keep the same file names, and differ only by extension. For example, an ASCII file containing the same data written to a dBase file might be named PF93193.DAT and PF93193.DBF respectively.

At a minimum, a good description of the data file must exist. This will greatly assist interpreting the data where the meaning of the eight-character description of the file in the filename is not clear to the user.

Most programs automatically append file extensions that indicate the type of file created. Some common extensions include:

- .DBF for dBase database files
- .DB for Paradox database files
- .Wkx for Lotus 123 worksheet files, where x is the software version
- .TXT, .PRN, .DAT and .RPT for ASCII files

.XLS for Microsoft Excel spreadsheets  
.mdb for Microsoft Access  
.sys for Systat files

Word processing files should be given a .WPx extension (where x is the version number for WordPerfect files) or .DOC extension for Microsoft Word files especially for documentation or final report files that are stored with other electronic records.

#### Data Exchange Formats

The standard database management software for the National Park Service and Denali National Park and Preserve has been dBase III Plus, which was required for Denali's Long-Term Ecological Monitoring Program. Recent developments in software design have greatly enhanced the portability of data between various platforms. Other database products, and even spreadsheet programs, can read and write files in dBase format. Word processors can directly read dBase files. As a result, the boundaries between products have grown less distinct and that trend will continue. The database of choice will be tied to a standard rather than a product. That standard is the xbase format. All final database files will be stored using a relational database software package and be available in dBase readable format. Spreadsheet programs have become extremely powerful and have a use for data manipulation. However, only true database programs have the features needed to ensure efficient use of data files through relations as well as put in place data entry controls that aid in quality control. Using other software is acceptable as long as files can be exchanged and is permanently stored in a relational database format.

Developments in multi-tasking, graphical user interfaces, object linking, multimedia, etc., indicate the park should move its computing activities to the Windows environment as hardware permits. The National Park Service recently adopted Microsoft Office as the standard on the Windows platform. With this action Microsoft Word has become the word-processing standard, Excel the spreadsheet standard, and Access the database standard. Excel and Access are both capable of importing and exporting in the aforementioned formats. Any changes in the database standard this may dictate will be made following evaluation of the new software standard.

### **COLLECTION AND DATA ENTRY**

Each data form should be accompanied by a document that describes the contents and codes of each field. Units of measurement and the meaning of coded categorical data are critical to document in the metadata. Definitions will be provided for each parameter collected in the dataset. Definitions will include the parameters (degrees Celsius, pH, etc.) as well as acceptable ranges of values. Such definitions can then be used to 'flag' possible outliers during the Q/A process.

A standard set of data entry forms should be compiled to ensure consistency across disciplines pertaining to the format of specific data fields and to provide for efficient computer data entry. Data forms for each discipline will be shared with the rest of the investigators to ensure each person's needs are being met. A consistent format will be developed where a core set of fields appears consistently on each form. Additional parameters particular to each discipline will vary across each form but will follow a consistent style. The forms should also be reviewed annually, prior to the field season, to incorporate and recommended changes to their format. Field procedures (protocols) must be archived with data to aid future users in interpreting data. For durability in the field and longevity for storage, the field form should be printed on 50 to 100 percent cotton fiber paper. Data should be recorded using a permanent ink pen. The Tombow rolling writer is easy to use in the field and has been found to exhibit a high level of performance when tested against other pens (Keay, 1992).

Alternatively, observations can be recorded directly onto magnetic media using portable computers or data loggers for automated or manual entry. In this case, a specially designed database or computer program is required. The following discussion assumes paper data forms will be used. Specific, step by step, instructions for data entry are included in Appendix D, page 22.

## Data Collected on Field Forms

Without proper preparation and following some simple guidelines for data entry the quality and integrity of the data will ultimately be debatable. There are three separate procedures needed to properly enter field data into a computerized form, to ensure these data are considered to be accurate.

Data Entry. This is the initial set of operations where data written on paper field forms are transcribed, or typed, into a computerized form (i.e., a database or spreadsheet). Where data were gathered and/or stored digitally in the field (e.g., on a datalogger), "data entry," in the context used here, is that stage where those data are transferred (downloaded) to a database in an office computer where they can be further manipulated. (see Appendix D, page 22)

Data Verification. Data verification immediately follows data entry and involves checking the accuracy of the computerized records against its original source, usually paper field records. While the goal of data entry is to achieve 100% correct entries, this is rarely accomplished; the "verification" phase checks the accuracy of all entries compared to the original source, and identifies and corrects any errors. Once the computerized data are verified as accurately reflecting the original field data, the paper forms can be archived and all further activities involving the data can be done via the computer alone. (see Appendix E, page 25)

Data Validation. Although data may be correctly transcribed from the original field forms (data entry and verification), they may not be accurate or logical. For example, finding a stream pH of 25.0 or a temperature of 95°C in a data file is illogical and certainly incorrect--whether or not it was properly transcribed from field forms. This process of reviewing computerized data for range and logic errors is the validation stage. This can be done during data verification *only* if the operator is intimately knowledgeable about the data. More often this will be a separate operation carried out by a project specialist *after* verification and with the goal of identifying both generic and specific kinds of errors in particular data types. Any corrections made to a dataset reflecting logic errors will also require returning to the original paper field records and making notations about how and (now) why those data were changed. (See Appendix F, page 27)

## Electronically Collected Data (data collected by data loggers)

Data downloaded from a data logger has its own set of procedures unique to the piece of equipment for downloading the data into the computer. Once the data has been transferred, the same procedures as outlined for transfer from field forms will be followed. The data stored in the module will not be cleared from memory until it is ascertained that the data was successfully transferred and a backup copy has been made of that data.

## Data from External Researchers

Data produced by researchers not employed by Denali National Park and Preserve will be transmitted to the National Park Service in the following manner:

Computer-recorded data: A copy of the data file(s) and supporting documentation on 3 ½ inch diskettes readable by IBM compatible computers will be forwarded. Data should be delimited and in ASCII or Standard Data Format (SDF), with the exception of files in dBase and Word format (list version in data.doc file described below).

In an ASCII file named "data.doc" the following information for each data file will be included with each submission:

- a) a written description of the data file contents, with sufficient detail to describe the contents of each file,
- b) the number of records per file,
- c) the number of rows of data per record,
- d) a listing of the data variables or fields in the data set including:
  - 1) variable number - if applicable
  - 2) variable name or label,
  - 3) variable type - numeric character (includes alphanumeric), date, logical, other,
  - 4) variable width and decimal - total number of spaces or columns occupied by the variable in the file and number of decimal places or columns,
  - 5) variable category labels or measurement unit - for all categorical variables list each category code followed by a category label; for all continuous metric variables list their respective measurement units,
  - 6) missing data codes - if applicable, code(s) used to denote missing data,
  - 7) any other information you feel is necessary to describe your data for future access and use.

**Note:** Many software packages have commands, which will print much of this information automatically (e.g. "List structure" in dBase, "display all" in SPSS).

Also, supporting non-data files, which are vital to the use of the data files, will be included. Examples include statistical package data description files (including compute and data transformation statements), index files, and programming files. A listing of these files, their content, and use in the "data.doc" file will be included.

Hand-recorded data: Documentation needs vary with type of data. Representative copies of all standardized forms, or if forms are not used, a description of the information that was collected will be included.

Instrumentation: Where instrumentation has been used to collect, record, or measure data parameters it is essential to record information about that equipment. The name of the instrument, manufacturer, model, date(s) of calibration, calibration method, etc. must be recorded for that equipment.

## DOCUMENTATION

Documentation of datasets is probably secondary in importance only to verification of the accuracy of the data in the files. Without informative and complete documentation, the content, quality, extent, known causes of variability and utility of the data remain unknown. The process of cleaning up undocumented data must begin with the first record, although even partially documented data give a data handler or project manager the ability to focus "cleaning" efforts on the sections of the data that actually need it.

### Software

Programming documentation is essential for any computer application that is written for use in a project. Documentation must be detailed enough that another programmer can amend the application at a later date without the assistance of the original programmer. Programming documentation includes a descriptive documentation file. Source code is adequately documented through the use of comments within the file. The source code should be organized by run order sequence, with each module clearly defined in the code. For each module, comments must include:

- 1) Module title.
- 2) Author's name and how he/she can be reached.
- 3) Module purpose
- 4) Input process (program), and output file names and notes

- 5) Any other notes appropriate to the module's data file management (erasure of temporary files, file naming and naming conventions, etc.).
- 6) Changes in the software. Keep records of added modules, modifications to modules, when, why added or modified, associated file and database changes, field changes. This should be kept as application documentation -- not just noted in the code.

### Data Dictionary

All resource data sets will be inventoried so that a listing of available information in the form of a data catalog can be found in a central location as well as be made available to a wider audience through the Internet, for example.

A data dictionary will identify standard data elements that will be used within each database. This document will provide detailed information about each data set; file structure, field lengths and types. The data dictionary will also describe each data element in an overall compilation. Information will include type of data (pH, sex, degrees centigrade, etc.), field name, length of field, type of field, and other pertinent information. Existing databases will be normalized to eliminate redundancies and inconsistencies in field names. All new data will conform to the standards outlined in the dictionary, thereby providing uniformity in data formats.

### Metadata

Good system documentation is essential and an on-going process. Documentation addresses the database as well as any custom applications that have been written to work with the data. This documentation includes information such as project name, principle investigator, how to contact that person(s) and a brief description of what type of data the file contains. It would list the structure of each field with a detailed explanation of the data within that field, including any codes that were used in recording the data. The documentation would identify known errors, missing data, or any other problems and indicate the current editing status of the data file.

All datasets will have basic metadata (data about the data) attached in a separate file. Digital geospatial data will comply with the FGDC content standards. Data finalized prior to January 1995 will meet the standards for "Legacy data". Data finalized after that date will comply with the full standards as outlined in the metadata standard, dated June 8, 1994. Depending whether the data is considered either spatial or non-spatial the metadata will follow a format similar to the template developed by the Federal Geographic Data Committee (FGDC) which addresses spatial data<sup>1</sup>. (See attached Sample Metadata Entry Form, Appendix A, p. 15) or the standard being developed by the Technology Transfer Division of the BRD for Non-spatial data. (Appendix B, page 20)

Documentation files should also contain the field data form. The park will develop and add to these procedures additions to the metadata standard that are needed to fit the situation at Denali. Appropriate metadata will be a required product of any activities that produce a new natural resource dataset.

## **QUALITY ASSURANCE**

### Editing Procedures

Datasets are rarely static; rather they often change due to additions, corrections, and improvement resulting from summary and analysis. The procedures to be followed when editing a dataset are outlined in Appendix G. These steps are to be followed when changing or editing data files, whether such change is to correct an existing dataset against written records, adding new records, or for changing its structure. The three caveats to this process are: 1) only make changes that improve the data while maintaining its integrity; 2) document everything that is done; and, 3) be prepared to recover from mistakes made during editing.

It is important that all data are validated as "truthful" and not misrepresentative given the circumstances and limitations of their collection, and that proper preparation for--and documentation of--all changes is made at the time editing is carried out. This procedure also requires that the user practice careful "version control" during editing to

ensure that changes are incremental, and that roll-back to a previous editing session is possible until such time as the file being changed is certified as correct and up to date. An example of final notes describing an edit is given in the appendix. (See Appendix G, page 31)

## **ACCESS**

Guidelines for the dissemination of datasets are given in Appendix H, page 38.

## **SECURITY**

The objective of proper data storage is to insure protection from loss and damage by such elements as temperature, or electric or magnetic forces. Care must be taken to properly store the data in all forms.

### Field Form Security

Completed field forms, as well as quality assurance documentation, should be filed in properly labeled file folders and stored in fireproof file cabinets. Program managers are responsible for file organization. Storage of these forms is to be arranged with the park's Research Administrator.

### Power

Electrical power at Denali is often subject to outages, surges, and brownouts. All computers within the Division of Research and Resource Preservation will be provided with surge protection and uninterruptible power supply equipment. Computers used by outside cooperators should have similar protection.

### Data Security

Backup files are the single most important safeguards against data loss. Backup copies of project files should be made regularly and religiously, and several people involved with the project should know where and how backups are stored.

Denali has recently integrated its computers into a local area network (LAN). This development will have great benefits for data management. It will allow data sets to be stored in a central location. The need for multiple copies of the same data thereby will be reduced. Tracking of versions will be made easier. It will be possible to backup files to a central source instead of maintaining individual backups for each computer. It will also allow access to the data to be controlled in a number of ways.

Database and documentation files, through all phases, should be backed-up on magnetic media. For short-term storage, streaming tape or 3 ½ inch floppies placed in cases or jackets that protect from bending and dust are appropriate. Files should also be stored on a computer hard drive. Eventually the network server will be the central storage facility for all on-line magnetic data. The backup tape sets will be stored in a fireproof safe or file cabinet with an additional set stored off-site. In this way, even in the event of a catastrophe that manages to destroy those tapes stored under fireproof conditions, one set will be safe in another location.

Placing a dataset on a central computer where a number of users have access presents inherent risks. The network operating system features access controls to the data through assignment of read/write permissions. The ability to make changes to datasets placed on the network must be strictly controlled and limited to the Principal Investigator or a person designated by that Investigator. When the decision is made to make a dataset available to others via the Internet, there will be another layer of security concern. It must be assumed that hackers will attempt to gain access to the local computer. Firewalls and other increased security measures must be in place to ensure this does not happen.

The preferred method may be to store the accessible data on a site removed from the 'official set'. Such a method is planned for the GIS dataset. The 'official' data is stored on-site with a mirrored copy in Anchorage. The accessible set will be stored on the NPS ftp site in Denver.

Hard copies of the latest editions of both the documentation and data files provide additional security and are useful, especially for editing and general information purposes. These will be stored in labeled file folders contained in a fireproof filing cabinet. Storage of this documentation is to be arranged with the park's Research Administrator.

### Backing Up Data Files

As more computers are brought on line with the network, backup will become more centralized. In the interim, the following procedures apply. Every computer user should have a procedure for normal system backup, usually on tape or diskette. However, data files may exist in various stages of editing seemingly from one moment to the next. For a certain time period they are massaged extensively, and for these reasons, normal system backup does not usually occur often enough to be sufficient. Normal system back up, however, does provide an additional and important level of security and should be used.

A daily backup of the files that have been changed during that day is important to ensure the security of recently changed files. At the end of the day running the daily backup utility will create a compressed ZIP file of that day's work. The file can be saved to floppy disk or the network drive.

The user with the internal tape drive and backup tapes will do weekly backup. Each backup will be a full backup of all the data that exists on that PC. Three tapes will be used to accomplish this. The user will cycle through the tapes on a weekly basis. Tape 1 is used the first week, tape 2 the second, and tape 3 the third. On the fourth week tape 1 will be overwritten and the process repeated.

On a quarterly basis the data manager will do a full system backup of each computer.

All final data collected as part of the Long-Term Ecological Monitoring Program should be stored on the hard drive of the following two computers:

- 1) Gateway Pentium Pro located in the Park Data Manager's office.
- 2) Network server located in the Computer Specialist's Office.

Park-based LTEM researchers and technicians who are not working directly on one of these computers should transfer their data (including documentation files) to them on a regular basis. The Division data manager will be responsible for ensuring this occurs and for protecting the data stored on these computers.

The LAN server undergoes a full system backup each night. Tapes are cycled through the server in such a manner as daily, weekly, monthly, and annual backups are maintained. The backups will be stored in a fireproof safe located above ground and away from the server. Maintaining a set off-site will still take place.

### Tracking Back-Up Files

Keeping track of many versions of many files that are edited irregularly can become confusing. A paper form can efficiently track recent versions. The form should include project investigator's name, data manager's name, and the name of each file. When a file is backed up, the date of the latest edition, what diskette that edition resides on, and who was responsible for that edition is entered on the form. (Appendix B, p. 20)

It is important that those individuals manipulating the data are careful to complete this form each time changes are made to a file. As long as the form is kept up to date it is a straightforward task to determine where the most current version of a file can be found as well as where to find earlier versions in the event they are needed.

As the network becomes more robust it will be possible to maintain a single version of a particular database that authorized users will have access to for editing. When the file is considered 'final' access will be limited to a 'read-only' basis and no further changes will be possible.

A final set of on-site and off-site tapes should be made at the end of each year to provide an annual record.

### Virus Protection

Computer viruses are capable of many things, including deletion of data or programs, modification of data, the introduction of typing errors, even reformatting of a hard disk (Skulason, 1992). To protect against such effects, care must be taken to prevent the introduction of viruses into the computer system. It is important that all diskettes to be used on a computer be scanned before use with a current version of some virus detection software.

Viruses spread by attachment to executable files or the boot sector of a diskette. Greatest protection will be afforded if only data and documentation files, and no executable files are placed in long-term storage on compact disks.

The Data Manager will check all computers within the Division on a monthly basis for viruses. F-prot is the virus-checking program used by the National Park Service. The program is available on the network and can be distributed for use by users who are connected or have laptops. If a virus is found, all diskettes on all computers will be checked. **Floppy disks received from outside the park will be checked for viruses before the files on them are copied to the PC.**

### **FINAL DATA AND ARCHIVES**

Final Data - The development, editing, updating, and final completion of a data set is the responsibility of the Principal Investigator who supervises the research or monitoring project. In stating that a particular set of data is final that investigator is certifying that all data for the specified period has been collected, proper verification procedures have been undertaken that ensure that the recorded data meets the designed quality standards, and the data is available for general distribution. All data not meeting these final standards will be identified as preliminary and labeled prominently with the appropriate disclaimers. It may be distributed on a case by case basis but should not receive general distribution

Sensitive Data - Certain data sets record information regarding resources that are considered to be of a sensitive nature. Information regarding these resources is considered sensitive because law either prohibits distributing it or the knowledge gained could be used to jeopardize the resource. Archeological resources are protected under the Archeological Resource Protection Act and providing specific locational information regarding these resources to the public is prohibited. The location of raptor nests could be of great interest to poachers or others whose actions could disturb the birds. A determination of the availability of such data will be made on a case by case basis.

### Freedom of Information Act

The Freedom of Information Act directs that most records held by the government be made accessible to the public, with few exceptions. Data held by Denali will be made available under the Freedom of Information Act unless it protected from disclosure by one of the following exemptions. Courts have held that if the information would risk the circumvention of statutes or agency regulations the data can be withheld. This finding has been used to restrict access to information regarding the location of threatened or endangered species and it would appear could be extended to other sensitive nest and den sites.<sup>1</sup> The other exemption applies to data if it is protected from disclosure under another statute (ARPA, Federal Cave Resources Protection Act, etc.). Requests should be evaluated on a

---

<sup>1</sup>: Memorandum from Office of the Solicitor re: Freedom of Information Act - requests for Location Data on Endangered Species, Aug 3, 1994

case-by-case basis. Questions regarding requests made under the Freedom of Information Act should be directed to the park Public Information Officer.

### Archiving

Final records become part of the park archives and are incorporated into that system. Data and supporting documentation will be accessioned and cataloged to the appropriate level of detail. Records (paper files as well as digital records) will be stored in the museum collection/archival storage. At that time a copy of the data will also be placed in the Denver Service Center - Technical Information Center (DSC-TIC).

Reports and records pertaining to the particular activity become part of the Division's central file system. They are filed using that system's method of organization. Central file records contain files from the last three years. Files four years and older are removed and incorporated into the park archives.

Hard drives and tapes are not sufficiently durable to insure long-term storage of important digital information. Diskettes formatted from one program may not be readable under another. At present it appears that the most feasible long-term storage media for computer files that is durable and unaffected by electrical and magnetic forces will be compact disk. The cost for the equipment to record to CD-ROM has dropped precipitously and is now within the reach of large, central offices. When data and documentation files, including the master documentation file, are completed and in final form, they will be stored on at least three compact disks. Each disk should be stored in a separate location in a fireproof safe or file cabinet. The hard copy of the master documentation files should identify the long-term storage location of each CD, including label and contents.

## APPENDICES

## Appendix A - Sample Spatial Metadata Entry

### Metadata

Identification Information

- \* Data Set Identity: Reno, NV-CA West 3-arcsecond DEM
- \*\*# Data Set Description: Digital elevation models (DEMs) are digital records of terrain elevations for ground positions at regularly spaced horizontal intervals. This data sets is one of a series available in 1- by 1-degree blocks, with a 3-arcsecond ground distance between each pair of digitized points. The series provides coverage for all of the contiguous United States, Hawaii, and limited portions of Alaska. The data were produced originally by the Defense Mapping Agency, and are distributed by the U.S. Geological Survey.

Theme

- \*\*# Theme Keyword Thesaurus: None
- \*\*# Theme Keyword: elevation
- Theme Keyword: altitude
- Theme Keyword: digital elevation model
- Theme Keyword: DEM
- Theme Keyword: hypsography
- Theme Keyword: topography

Bounding Coordinates

- \* West Bounding Coordinate: -119.0
- \* East Bounding Coordinate: -118.0
- \* North Bounding Coordinate: 40.0
- \* South Bounding Coordinate: 39.0

Data Quality Information

- \*? Beginning Date of Information Content: 1973
- Thematic Quality
  - \*? Thematic Accuracy Report: [agency statement]
  - \*? Logical Consistency Report: [agency statement]
  - \*? Completeness Report: [agency statement]
- Positional Quality
  - Horizontal Positional Quality
    - \*? Horizontal Positional Accuracy Report: [agency statement]
  - Vertical Positional Quality
    - \*? Vertical Positional Accuracy Report: [agency statement]

Spatial Data Organization Information

- \*\*# Direct Spatial Reference Method: Raster

Spatial Reference Information

Horizontal Coordinate System Definition

Geographic

- \*\*# Latitude Resolution: 3.0
- \*\*# Longitude Resolution: 3.0
- \*\*# Geographic Coordinate Units: Decimal seconds

Geodetic Model

- \*\*# Ellipsoid Name: World Geodetic System 72
- \*\*# Semi-major Axis: 6378135.
- \*\*# Denominator of Flattening Ratio: 298.26

\* - places where a data entry is required. Repetitions are not so noted.  
# - places where a 'boilerplate' entry for a series of data can be used.  
? - places where a 'boilerplate' entry for a series of data might be used.

Vertical Coordinate System Definition  
 Altitude System Definition  
 \* Altitude Datum Name: National Geodetic Vertical Datum of 1929  
 \*# Altitude Resolution: 1.  
 \*# Altitude Distance Units: meters  
 \*# Altitude Encoding Method: Implicit coordinate

Status Information  
 \*# Data Set Status: Available  
 \*# Release Date: 1987  
 \*# Maintenance and Update Frequency: Irregular

Lineage =

Source Information  
 \*? Source Citation: various cartographic and photogrammetric sources  
 \*? Source Citation Abbreviation: DMA1  
 \*? Beginning Date of Source Currentness: Unknown  
 \*? Source Contribution: elevation information

Source Information  
 Source Citation: Defense Mapping Agency, no date, Reno, NV-CA West Digital Terrain Elevation Data.  
 Source Citation Abbreviation: DMA2  
 Beginning Date of Source Currentness: Unknown  
 Source Contribution: data for reformatting to DEM format

Process Step  
 \*? Process Description: The 1-degree digital elevation data models were produced by the Defense Mapping Agency (DMA) using cartographic and photogrammetric sources. Elevation data from cartographic sources are collected from any map series (1:24,000- through 1:250,000-scale). The hypsographic features (contours, drain lines, ridge lines, lakes, and spot elevations) were digitized and processed into the required matrix form and interval spacing. Elevation data from photographic sources were collected using manual and automated correlation techniques. Elevations along a profile were collected at 80 to 100 percent of the eventual point spacing. The raw elevation were weighted with additional information such as drain, ridge, water and spot heights during the resampling process in which final elevations were determined for the required matrix form and interval spacing.  
 \*? Source Used Citation Abbreviation: DMA1  
 \*? Process Date: Unknown  
 \*? Source Generated Citation Abbreviation: DMA2

Process Step  
 Process Description: The digital elevation model data were provided by the DMA to the U.S. Geological Survey (USGS) in the DMA Digital Terrain Elevation Data Level 1 format. The USGS reformatted the data to the DEM format for distribution to the public.  
 Source Used Citation Abbreviation: DMA2  
 Process Date: Unknown

Entity and Attribute Information  
 Entity Type  
 \*# Entity Type Label: elevation point  
 \*# Entity Type Definition: a point of known elevation.  
 \*# Entity Type Definition Source: None.

Attribute

\* - places where a data entry is required. Repetitions are not so noted.  
 # - places where a 'boilerplate' entry for a series of data can be used.  
 ? - places where a 'boilerplate' entry for a series of data might be used.

\*# Attribute Label: elevation  
 \*# Attribute Definition: altitude above or below a reference datum.  
 \*# Attribute Definition Source: None.  
 Attribute Domain Values  
     Range Domain  
 \*        Range Domain Minimum: 999.  
 \*        Range Domain Maximum: 2641.  
 \*# Attribute Units of Measure: meters  
 Distribution Information  
     Distribution Contact  
         Contact Information  
             Contact Organization Primary  
 \*#            Contact Organization: Customer Services  
 \*#            Contact Mail Address: EROS Data Center, Sioux Falls SD 57198  
 \*#            Distribution Liability: [agency statement]  
 Standard Transfer Option  
     Digital Form  
         Digital Transfer Information  
 \*#            Format Name: DEM  
         Digital Transfer Options  
             OMine Options  
 \*#                Offline Media: 9-track tape  
                 Recording Capacity  
 \*#                    Recording Density: 1600  
                     Recording Density: 6250  
 \*#                    Recording Density Units: characters per inch  
 \*#                    Recording Format: ASCII recording mode; available with no internal  
                     labels or with ANSI standards labels; the logical record length is 1024  
                     bytes; the block size is a multiple of 1024 up to 31744 bytes  
 \*#                    Compatibility Information: None  
 \*#            Fees: Purchased separately: \$40 per file; purchased in groups of 2 to 6: \$20 per file; purchased in  
             groups of 7 or more: \$90 base fee plus \$7 per file.  
 Metadata Reference Information  
 \*    Metadata Date: 19940404  
 \*#    Metadata Standard Name: FGDC Content Standards for Digital Spatial Metadata  
 \*#    Metadata Standard Version: 19940331

\* - places where a data entry is required. Repetitions are not so noted.  
 # - places where a 'boilerplate' entry for a series of data can be used.  
 ? - places where a 'boilerplate' entry for a series of data might be used.

Under Servicewide Development



## Appendix D - Data Entry Procedures

This is really a simple monotone process, which is easy to perform and follow. It is NOT a trivial operation, however, because the value of the data are determined by their accuracy--which is still unconfirmed and at risk at this stage of computerization. Remember that the single goal of data entry is to *transcribe* the data from paper records into the computer with 100% accuracy. Proper preparation and adherence to the steps below guards against having to do the same work over again during data verification. But do relax; making some errors is nearly unavoidable when entering lots of data, and these errors will be dealt with during the verification phase.

1. Have two people available for entering data. Although not required, experience has shown that having one person read the data off the paper and another enter it into the computer will make the work go much faster and result in a much lower error rate. If another person is not available, the person entering the data should try to work a bit slower (=not rush) to compensate. Like many monotonous tasks there is a rhythm to be discovered in data entry, whether doing it alone or with another, and setting a reasonable tempo is important for avoiding mistakes or getting distracted; it may also make the job more pleasant.
2. Prepare your workspace. Do NOT begin doing data entry in a messy setting. Clean a space on your desktop near the computer and do not allow other "items" to arrive and clutter that space. You want to proceed cleanly through the process and do not need distractions that will make you lose your place and have to start over. You will likely be working with two piles of paper documents, one from which you get un-entered data and another where you put ones you've completed, so think about where they will go. You will also want to have a notebook or pad of paper handy for any notes you may want to make, and a few fine color markers (green, blue and red).
3. Examine the documents needing transcription. Become familiar with the data forms, the differences in people's handwriting, and whether or not some "fields" are intentionally left empty on some of the forms. Some errors and/or omissions are detectable or may be suspected at this stage and may necessitate setting aside some of the forms for data clarification or correction by the field staff *before* attempting to enter them into the computer. Also identify in the documents what constitutes a good stopping point, since interruptions (or the end of the work day) are likely. Most would agree that the best stopping point is when a single, complete field form has been entered; avoid stopping in the middle of a logically single operation.
4. Prepare the computer. Each dataset may have its own, unique data entry procedure on the computer. The specific application program and/or file to use should be provided by the project manager and be consistent with the data content and data structure policies of the I&M Unit. Almost always, data are entered into an empty, "fresh" database to avoid contaminating existing data [new data are appended to the master data only after verification/validation].

If you are unfamiliar with the program or application to be used for entering the records, spend a little time practicing first. The best entry applications will also do some validation of data on-the-fly, such as checking for valid ranges, dates or spelling, and warn the typist as errors are made and provide the opportunity for correction before the data are committed to a file. You need to know how to commit both a "field" entry and a complete record (e.g., often TAB is used to move between fields, and ENTER is reserved for committing the entire record [whether you finished or not!]). Also, know how to correct mistakes you know your fingers are making while typing (i.e., the effects of using backspace, back-tab, del). Once you know where and how to enter the data you are ready to move on.

5. Enter the data, one logical "set" at a time. Here you simply enter the data one logical data "sheet" at a time (usually one complete field form). Record in your notebook errors you know you've made or any questions that arise about the data content; these will be useful during data verification. Initial each paper form as it is completed to avoid confusion about what has been entered and what has not (green is a positive color to use). **Interrupt your**

**data entry only at logical stopping points.** If you reach stopping points, make a working backup copy of the data for safety's sake if your software does not do so automatically. When you have completed the entry of all records for this dataset, **Immediately continue through the next three steps--this is not a reasonable stopping point!**

6. Print a copy of the computerized data. Print a copy of all the data you entered for the verification stage (see the project manager for formatting details if you are unsure). Do **not** apply any sorting to the file, since it must be checked in the same "order" as it was entered to speed verification from matching field forms. Check to make sure **all** the data were printed (none are "off" the right margin) and are readable (font size and attributes), since this printout will be used for data verification. Do NOT prepare any reports from these data at this stage since they have not been checked and are almost assured of having some errors. If you were cautious during data entry, the number of errors will be few and the verification process will proceed much more quickly than the entry phase.

7. Initial and date the original records and the printed copy. Indicate on a cover sheet or other suitable location that these data were entered; provide full name, or initials, and a date. Record identical information at the top of the printout of the entered data. Keep the original and the printout together for use in data verification.

8. Make and store a backup copy of the data. To avoid data loss, immediately make a backup copy of the data just entered and store it in a second location (or provide a copy for safekeeping to the project or data manager). Congratulations! The data entry phase is now complete.

## Appendix E - Data Verification Procedures

The verification phase is carried out to ensure that all the data were entered and that they were accurately transcribed.

1. Two people are best here, too. If at all possible, use two people for the verification phase. This process involves two sets of paper simultaneously and goes much more rapidly than data entry, and thus generates a greater opportunity for confusion or losing one's place when working alone. Verification is best accomplished with one person reading the original data sheets (the "reader") and the second comparing with the same data on the printout (the "checker"). Until the computer is trained to read the data back by itself (a working prototype has already been demonstrated for ASCII data!), two people are better (=more accurate, faster) than one. We will assume hereafter that a pair of people are working together.
2. Prepare your work space. As with data entry, a clean work space will promote better control of the verification process. Both the reader and checker should have space for their checked and unchecked pages on the desktop. The checker will need a red and green fine-tipped marker for identifying errors and indicating corrections have been made; the reader will need a marker, too (green is good). Rulers should be provided for easy reading and maintaining a position or advancing one record at a time on the printed page. Keep your notebook handy for recording any other notes that might be useful during validation.
3. Compare data and note differences on printout. The reader reads the original data (field forms) and the checker compares that against the printout made after data entry. The three common types of error that will be found are duplicated records (entered twice), missing records (inadvertently skipped during entry) and misspellings (wrong number or code). The checker controls the speed of the reader, and halts the reader when any discrepancy is found. When an error in the printout (=computerized records) is found, the **correction to be made is noted in red on the printout; no correction marks should be made on the original data sheets at this point.** After verifying the data from each field sheet the reader should date and initial the original field form at the top (or where provided) stating that verification was done. Continue the reading and checking until all the data sheets for that dataset have been compared. Now you have an original set of data sheets with completion marks (both entry and verification), and a set of printouts with corrections needed marked in red.
4. Correct identified errors in the computer files. Return to the application used for data entry (or one provided specifically for editing) and correct the errors as indicated on the printout. Make each correction separately (i.e., avoid doing a "search and replace" that might have unexpected consequences). As each correction is made, the red mark on the printout should be "OK'd" with green. Continue until all identified errors are corrected in the computerized file, and check the printout again for any that were missed (red without green check). Finally, date and initial the printout at the top that all errors were corrected. Save the printout with the original field form, as it serves as direct evidence of the completion of entry and verification.
5. Perform simple summary analyses. Use the computer to generate some simple summary statistics for the entered data. This is important because even when care is taken up to this point, it is possible to have missed a duplicate or omitted record. For example, do a count of elements that are known to be constant, such as the number of sites sampled, plots per site or dates per sample. Be creative by asking the same question in different ways; differences in the answer provide clues to where errors reside. The more checks you can devise to test the completeness of the data the more confidence you will have that the data are completely verified.
6. Make and store a backup of the data. Make a copy of the verified data file(s) and store it where instructed. Pass a second copy of the file(s) to the project and data managers with appropriate documentation. Make a copy of the original field forms. Attach the printout to the original field form and store in the specified area; put the copy of

the original form in a second location. Check with the project manager for the exact file cabinets to use if unsure about storage locations.

## Appendix F - Data Validation Strategies

Unfortunately, there are no step-by-step instructions possible for data validation, because it might be considered more an art than a *standardized* quality control procedure. Nonetheless, it is a critically important step in the certification of the data. Invalid data commonly consist of slightly misspelled species names or site codes, the wrong date, or out-of-range errors in parameters having well-defined limits (e.g., elevation). But more interesting, and often cryptic, errors are detected as unreasonable metrics (e.g., stream temperature of 70EC) or impossible associations (e.g., a tree 2 feet in diameter and only 3 feet high). We have come to call all these types of erroneous data "logic errors" since using them produces illogical (and incorrect) results. The active discovery of logic errors also has direct, positive consequences for data quality and provides important feedback to the methods and data forms used in the field. Validation, therefore, is not a step to be ignored until after statistical analyses reveal problems with the data.

Wherever possible the data entry application should be programmed to do the initial validation. The simplest validation to perform during data entry is range checking. This includes such actions as ensuring that a user attempting to enter a pH of 20.0 gets a warning and the opportunity to enter a correct value between 1.0 and 14.0 (or better yet, within a narrow range appropriate to the study area. Not all fields, however, will have appropriate ranges known in advance, so knowledge of what is "reasonable" data and a separate, interactive validation stage is still important. The data entry application should also use "pop-up pick lists" for any standardized "written" items where spelling errors can occur. For example, rather than typing in a species name (where a misspelling can generate a "new" species in the database), the name should be selected from a list of valid species, and "picked" for automatic entry into the species field. Again, not all written fields can use a list, but where they can be used they should be.

One of the most important activities of rigorous validation, however, is to return to the original data sheet (*and* the printout and 2nd copy) to make corrections and notations about the errors that were found and fixed in the computer files. Without annotating the original field forms, the computerized and paper records are out of synch. If this is discovered without adequate documentation explaining the differences, *all* of the data are rendered suspect. This is so important that it really needs to be repeated more strongly:

**WHEN VALIDATION ERRORS ARE FOUND IN THE ORIGINAL DATA, BOTH THE COMPUTER FILES AND THE ORIGINAL FIELD RECORDS SHOULD BE CORRECTED. Only when the original forms are annotated with the same corrections will the correspondence between computerized files and field forms be kept exact. Failure to correct the original field data forms will create havoc and doubt about the integrity of the data if it is later discovered that the field data and the computerized data do not match. And, don't forget to make the same correction notations on the other copies of the field forms.**

The following generic suggestions can help build a validation strategy for any dataset, and examples of approaches we've discovered (and strange errors) are also provided.

a. Catalog the error types found in each dataset. Once particular validation errors are found, it is important to catalog them for that dataset. The notes on the error should include a description, how it was detected, and how it was corrected. Both simple, generic errors and more esoteric and cryptic errors need to be documented. This catalog of errors will be a valuable reference for the next validation session, and ultimately for building formal validation procedures into the data entry process and other automatic, post-processing error checking routines.

b. Perform exploratory data analysis to look for outliers. Database, graphic and statistical tools can be used for ad-hoc queries and displays of the data. Such exploratory techniques will help identify obvious outliers. Some of these may appear quite unusual but prove to be valid, and thus at least need to be confirmed. Noting these "true" unusual values in documentation of the dataset also saves users the trouble of attempting to perform the same confirmation themselves.

c. Modify the field data forms to avoid common mistakes. With a catalog of errors and some exploratory data results in hand, it makes sense to reevaluate the field data forms as the source of the logic errors. Often minor changes (or small font annotations) to a field form will remove any ambiguity regarding what to enter on spaces in the form. In fact, any time the same type of validation errors occur repeatedly in different datasets, it is usually the field form that is at fault, not the field crew. Of course, it could also mean that the protocol or the field training is faulty, both of which are also important to recognize.

d. Examples of more insidious validation discoveries. Below are several examples of logic errors discovered in datasets. These should all be interesting and informative to the active data explorer. They demonstrate how errors can hide, and some generic and specific approaches to finding them. The take-home lesson really is: Seek and ye' shall find. Looking for errors in a data file, however, is probably not a career option; you're also supposed to do other work. At some point, then, you must stop searching for problems and accept the data as certifiably verified and validated. Proper data analysis will ultimately (and hopefully?) reveal any remaining errors in the files. Keep in mind that the most effective mechanism for avoiding spending time on validation is to get the right data into the computer in the first place. This would include having a complete set of SOP's and protocols for quality control: intelligible and unambiguous field methodologies, a well-trained and tested field staff, well-organized field forms, and data entry applications with simple validation built-in. Last but not least, it should be noted that exploring the data looking for logic errors is also a good way to "get to know" the data intimately; actually finding real errors means you're beginning to see what is true.

Wrong year. A simple typo during data entry creates a logical "set" of data for a year in which samples were never taken (the same can happen for month). This can become cryptic if the data are sorted by date before verification (not a good idea), thus the entry moves away from its true neighbors. If sorting creates the appearance of "missing data" where those records should have been, the appropriate corrective action during verification might actually *create* duplicate records in the file rather than fix the ones that were wrong--leaving two problems instead of one. Even when left in its original order, however, this error might go undetected because checkers can sometimes "see" what the readers say--especially when the month and date become the items of focus and the single year digit is not examined. A summary analysis reporting the count of total records for the dataset will also be correct. Summary explorations for the number of dates per year, or the number of samples per year will detect this kind of error by revealing a "year" that didn't belong, and the rest of those data records reveal which ones need correction. Identifying "wrong site code" errors follows an identical process, where a slight mis-typing was not visually identified during verification.

Wild temperatures. Stream temperatures can show really wild variation and yet be completely verifiable and valid. For example, some older data, or the occasional spurious recent record, may have been taken in Fahrenheit rather than Celsius. There is a big difference, obviously. This is really a protocol problem and not a data question, but where quality control procedures during data collection were lax, these types of errors are often found only during data validation or (more annoyingly) analysis. Routinely producing a box-plot or histogram of numerical data will reveal dramatic outliers, and when the original data forms are consulted, true outliers vs. errors in measurement scale or units become apparent, as does the correction for the files (convert the measurement to the appropriate units).

Trees that shrink. Analysis of data from a vegetation monitoring program that included remeasuring trees at permanent plots every five years found that some of these remeasured trees were getting smaller--recent DBH's (diameter at breast height) were less than the original measurements five years before. Tree trunks of live trees don't get smaller. Some serious detective work revealed that the data were entered accurately (verifiable), but that there appeared to be slight-to-moderate differences in the accuracy and exact methodology used by current vs. previous crews. A "Search and Compare" program was then written to parse the data and identify and scale the differences

between trees, revealing the extent of the "damage" in the data. This, unfortunately, was not a problem that could be fixed by editing the data files. Rather, it revealed a previous violation of protocol standards resulting in data of poor quality and rendered useless for their original purpose.

## Appendix G - Guidelines for Data File Editing

### 1. Before Editing

- a. Have a Notebook ready to record your actions.

The user should have a notebook to write any and all notes about what is done to a file. These working notes may or may not become part of the permanent record for the data, but are necessary for reconstructing the strategy used to change a file during an editing session. Whether or not these working notes are saved, be forewarned that a formal written summary and explanation *will be created*. This will be from each editing session, including a listing of all changes made, when they were made and by whom; this editing summary will become part of the documentation permanently associated with the data from that point forward. Also, your notes may later prove invaluable as a guide for accomplishing specific editing tasks in subsequent sessions.

- b. Prepare a clean work space on your computer.

It is best to work on files in a safe place on your computer--away from other files that might accidentally get altered or deleted. Using an \EDIT or \WORK directory for this purpose is recommended. If you do a lot of editing, working in the same directory will become a habit. It is probably also useful to have a \VERSION subdirectory in your work space to store numbered versions of your working file if the current directory becomes crowded. In that case, the main working directory might only contain the current copies of the files undergoing change, while the roll-back versions will be safely tucked away.

- c. Work only on copies of files, and one-at-a-time if possible.

Never work on your only copy (the original) of a file. Make a working copy of the file, and even better, give it a slightly different name. Choosing a shortened name for your working copy can facilitate loading, saving and version control procedures. For example, if you are working with the file STG3GC.DBF you may want to shorten the name during the editing session to Sn.DBF, where the 'n' is the version number. If two or more files are edited simultaneously (i.e., if they are relational), use similarly coded version numbers to remind yourself that they are both at the same stage of editing.

- d. Work on a subset of the data whenever possible.

Here you want to avoid corrupting any data that wasn't to be edited. For example, if a field named SITECODE needs to be adjusted for only one year in a multi-year file, it is best to isolate records for that one year before any editing begins. This can be done by splitting the original file into two parts, editing the one that needs it, and then recombining them later. (Remember to track the whole file and separate pieces in your notes.) Another approach is to use a "filter" tool in a database program to allow access only to the records you want. In either case, you will prevent accidental changes to data in non-target parts of a file.

- e. Define your editing strategy in writing.

1. Establish working file information.

Write down the names of the files to be worked on, their initial dates, sizes and the number of records. If you will be renaming a copy of the file to work on, record the working name for the file that includes an initial version number (e.g., "S0.DBF is the original SITE.DBF file."). If files are to be edited in more than one format (e.g., as both ASCII and DBF) to take advantage of

different editing tools, state so at the outset. Include information about the file extensions that will identify them (i.e., "ASCII versions of the same-name DBF files will have an ASC extension.").

2. List what changes need to be made, and sequence them.

Before beginning a data editing session, write down precisely, in as much detail as possible, what is about to be accomplished. If several steps are known to be needed to "fix" a file, they should be written down separately and examined carefully *before any editing begins* to evaluate whether any one change might adversely affect later steps. This examination may also reveal how to arrange a cascade of changes to be most efficient. If order of actions is important, explain why in your notes before beginning.

A common example of poor planning is using global search & replace functions indiscriminately, such as wanting to increment a few numbers by one. You might start by changing all "1" to "2", and quickly discover you can't distinguish the original two's anymore and all 1's in compound numbers and other codes were also changed to 2's--not what was intended. In that example you would have recorded beforehand the proper strategy of replacing the highest number first, and then working downward, and restricting your edit to the fields where they were needed.

3. Define the tools you will use for editing.

Name the computer programs and/or file editors that will be used to make changes to a file. This needs to be stated only once in your notes for the editing session. If you perform some unusual feat of magic to accomplish a difficult task, or simply try some advanced feature out for the first time, you may want to be recording the steps you take in detail. ("Wow! Perfect! Now how did I do that?!")

## 2. During Editing

- a. Keep making notes.

If you have properly planned and documented your strategy for the editing session, there is little left but to carry it out. Remember to record each step in the edit process, including names and versions of files--especially noting any changes in the number of records as a result of a change. It is best to use numbers in your notes about each step of the edit. You can improve your version tracking by using the step number as the version number when each step is completed (e.g., S2.DBF is the *result* of step 2 in your notes of the editing process).

Save your work often.

Save your files often, using timed-autosave features of the software you are using if it is available. The interval you choose for saving should be flexible and is not necessarily the same as a renaming of the file as a new version. For example, if you have a small series of relatively simple search & replace operations to carry out, you may want to complete them all before saving since it would be an easy task to do it again; at that point you may have completed a new version of the file and rename it so upon saving. However, if your step in the editing process requires many changes to individual records, you may want to save at, say, five minute intervals or even after each record is completed. In the latter case, keep recording in your notes or on scrap paper the last record edited when you last saved.

- b. Track your versions of the edited file.

Be sure to keep all intermediate, numbered versions until the full edit is complete, using compression software for their storage if disk space is limited.

### 3. After Editing

- a. Backup your new file and the version series.

Immediately make a copy of the "fixed" file and its intermediate versions to diskette, even if only for temporary storage. You don't want a power outage or hard disk crash to nullify your good efforts.

- b. Review your pre-edit notes.

Check off the changes you wanted to make against the notes you made during editing to double-check that all changes were made.

- c. Formally document the changes made.

Here you want to create a formal statement of what was done to the file. Include your name, the date, the file(s) that were changed and a concise list of the changes that were made and why, and the version series used during the edit. Each record that was changed does not have to be listed if the change applied was more or less global in nature, but if individual records were separately adjusted for a particular reason (i.e., to correct an error) they should be identified individually. The documents that accompany the file--including these edit summaries--should detail the entire history of the file, so don't leave anything out, no matter how minor you think the change was (for example, changing a single date in a 20,000 record file still needs an explanation). Any changes to a long-term database should be considered non-trivial.

- d. Archive the edited file and associated documentation.

When the edit and documentation are complete, follow the guidelines for formal archiving of the file and its associated documentation. In brief, this means:

- I. Print the edit session report.

A paper copy of the formal edit session documentation should go into a folder associated with that file or project;

- ii. Print the file, if required.

Optionally, a printout of the final version of the edited file should be made (if that has been the standard procedure for that project);

- iii. Archive and catalog the file(s) and documentation.

The final version of the file, computerized copies of the edit information, and any previous documentation associated with the file are then (re)catalogued and archived to safe, off-site storage.

- e. Update all computers that need the new file version.

Copies of the new file(s) now need to be distributed to those computers where they are used. This is a process best handled by the individual responsible for the project or the data. The project manager should always have a current listing of the current date, size and number of records in the file

- f. Update any Master record listing current data files.

If a Master record is kept that notes the current version of any file, it should be updated. This could be a notebook to which all users have access to check on the "current status" of their data, or a computerized database used for the same purpose. In any case, if there is a place to write down that the file is now "newer" than it was before, do it immediately.

DATA EDIT REPORT (EXAMPLE)

**Name:** Steve Tessler

**Date:** October 7, 1993 (completion date)

**File(s) Edited:** AQINS.DBF from Aquatic LTEMs project

**Reason for Edit:** Taxonomic code changes required to the TAXA field as per correspondence with Steve Hiner/Reese Voshell @ VPI. Changes needed and taxa-change categories are fully outlined in the correspondence and "Change Sheets" created for this edit, and include changing certain generic ID's to species, eliminating "terrestrials" and "impossible" taxa (for Virginia) in the data, and regrouping *Baetis* and *Pseudocloeon* as *Baetis* complex.

**Program(s) Used:** FoxPro 2.5 and DBBrowse for DBF files; TSE pre-release 1.0 and QEdit 2.15 (both Semware) for ASCII files.

**Original File Information:** AQINS.DBF, 13,779 records, contains data from 1986 through 1992. Full error-checking pending.

**Final File Information:** AQINS.DBF is now version 7 of this edit, dated 10/07/93, time 10:15:23.03a, with 13,731 records.

**Editing Details:**

1. Created AQINS.1 as a tab and "" delimited ASCII version of the original DBF. Files with a number extension are ASCII.
2. Did global delete of terrestrials and "impossibles." 33 records removed; new number of records = 13,746. Saved as AQINS.2.
3. Found error while previewing for *Peltoperla* changes. 15 lines were duplicates in 3L301 2nd Qtr 1988; confirmed by checking paper records; they were deleted. Saved as AQINS.3, #Rec now = 13,731.
4. Made unusual, one-time-only changes as per Change Sheets. 19 records were changed; # rec still 13,731. Saved as AQINS.4.
5. Changed all UNID to straight taxon code (i.e., removed X's from the code). 138 X's removed, no rec# change. Saved as AQINS.5.
6. Changed genus to species for monotypic genera and *Perlesta* to *P. placida* group as per Change Sheets. 395 changes made to 15 taxa. Saved as AQINS.6, still with 13,731 records.
7. Changed names up one taxon level globally as per Change Sheets. 358 changes made still 13,731 records. Saved as AQINS.7; loaded into AQINS7.DBF for checking and sorting. Re-saved as AQINS.DBF, final version. Edit complete.

## Appendix H - Guidelines for Disseminating Data

### Introduction

These guidelines are currently being developed with regard to preparing and sending data to those academic cooperators who are analyzing data for us. When the need arises to distribute data to a wider audience, some of the specifics of the procedures described below may need to be modified (such as only providing datasets in a standard delimited ASCII format rather than customizing datasets for individuals).

As we get more fully involved in the archiving of datasets and building of a metadata catalog, some of the operations described individually below will already have been done for each dataset. So, the items described below necessarily overlap the guidelines for proper data archiving at this time. When all of our data are fully validated, archived and catalogued, they will already exist in a form easily transferred without additional preparation.

The following steps should be carried out, in the order listed, to facilitate a transfer of data and supporting documentation that fully describes the data extent and limitations. An example data file description is given at the end of these guidelines.

### Items That Precede Data Preparation

1. Identify the data recipient.

Identify the target individual who will actually work with or process the files to be sent; i.e., with the consent of the principal investigator, work out the transfer with the data manager who will actually receive, handle and process the data. The next four items are addressed through discussion with that data recipient.

2. Identify the data to transfer.

Determine whether the entire dataset or only a subset of the data needs to be prepared for transfer. The recipient should have a good idea of what is in the dataset; faxing a copy of the data structure and a descriptive text before this discussion is a good idea.

Determine the format for the data.

Determine the data format needed by the data recipient (ASCII, DBF, spreadsheet, etc.). If ASCII files are desirable, then determine if the data should be in a fixed-column format or delimited by a specific character. Since ASCII conversions of DBF files from DBMS's often use quotes on character fields but not on other types, determine if this is acceptable or if all fields should either be quoted or unquoted.

3. Determine the format for accompanying documentation.

Determine any requirements for the format of accompanying documentation (e.g., ASCII w/ 65 column maximum width, WordPerfect 5.0 but not greater, etc.).

4. Determine the transfer medium and method.

Determine how the transfer will take place (e.g., 3.5" disks via U.S. mail, FTP to a specific Internet address, etc.), and whether file compression of a group of files is acceptable. Since most data files compress very well, transfer of compressed, self-extracting archives should be encouraged in all cases.

#### Data Preparation Procedures

1. List the data items, in detail, to be prepared.
2. Prepare the data.

Prepare the data to be transferred in accordance with the agreed upon formats. Fully document on paper any modifications, additions or changes in structure that are carried out on the files during this process. For example, the recipient may only want selected fields from a database. During preparation of this data, record the actions taken to achieve this goal.

#### Data Documentation Procedures

1. Document the individual items to be transferred:

1. List of files.

List all file names, dates, sizes and any directories and subdirectories that are to be transferred. Write brief, informative descriptions of each item.

2. File relationships.

Describe any relationships between individual data files. For example, if the data are relational and fully normalized, identify the primary and foreign keys in individual files used for linkages. Include a text diagram showing these relationships whenever possible.

3. For each data file, prepare:

- (1) A table showing the data file structure:
  - (1) the total number of records
  - (2) the size of each record
  - (3) the number of fields per record
  - (4) the names of fields, in record order
  - (5) field type, size, etc.

(6) a description of the field

(2) State how missing values are coded

2. Describe the full dataset, including limitations.

Document the dataset with a descriptive paragraph and a disclaimer if necessary; separate comments may need to be made regarding each individual file. Be sure to explain any known problems with the dataset--such as different protocols followed over different years, changes in equipment, detection limits or resolution, etc. Briefly describe the project that created the dataset, acknowledge contributors and provide an acknowledgment of the funding source.

#### Assemble The Transfer Materials

1. Review.

Review all of the items above, especially to check that any descriptive documents are also catalogued (as in noted above).

B. Double-check that complete documentation is provided.

Both printouts and on-disk copies of all descriptive documentation material will accompany the dataset. (Printouts are obviously not sent via e-mail.)

2. Compress the dataset, if required.

If compression was agreed upon for data transfer, compress the file collection with any subdirectories intact, and make it self-extracting.

3. Populate the transfer medium.

Prepare the distribution medium and populate it with the collection of files to be transferred. Check again with the list made above to be sure all items are included that need to be.

4. Prepare a cover sheet.

Write a brief, dated cover sheet addressed to the data recipient, with a return address, phone and fax numbers. If the data are compressed on the transfer medium, write a brief description of how to decompress the files, and how much space they will take up when decompressed.

5. Assemble the printed documentation.

Assemble the printed copies of all documentation that will accompany the data, with the cover sheet on top.

### Archive And Document What Is Being Sent

For our records, make a compressed version of the collection of data file(s) and accompanying material. A copy of this compressed collection should be backed up to a storage disk as a record of what was transferred.

### Send The Data

The next two pages show a sample of a descriptive text file for a dataset.

**SAMPLE OF DESCRIPTIVE TEXT ACCOMPANYING A DATASET**

AQINS.HDR -- description of AQINS.TXT (ASCII form of AQINS.DBF)

=====  
 AQINS.DBF and AQINS.TXT contain 13,731 records from 8/7/86 to 9/28/92

This is a brief header definition and description of the AQINS.TXT file containing the fixed-column ASCII version of AQINS.DBF. There are 8 fields (variables) in the file we are sending you for each record (row or line). There are a total of 13,731 records in the file. Each record is based on the actual presence of a single taxon in a specific sample. The following notes detail specifics of the AQINS.TXT file and describe how it differs from the original DBF file.

Description of Fields in the original AQINS.DBF dBase file (see LTEMs manual).

NAME	TYPE	LEN	DESCRIPTION
1. SITE	Character	5	LTEMs Site code
2. QUARTER	Character	1	Annual quarter (1-4)
3. SMPDATE	Date	8	Sample Date
4. ASMPL	Character	1	Sample # for a method
5. SMPMETH	Character	3	Sample Method
6. TAXA	Character	7	Taxon code
7. INSCNT	Character	4	Taxon count in the sample
8. METERCNT	Character	5	Calculated #/sq.meter
9. WEIGHT	Character	6	Individual weight
10. STAGE	Character	1	Individual stage
11. FUNCGRP	Character	2	Taxon functional group
12. REARED	Logical (T/F)	1	Individual reared or not

First, the fields METERCNT, WEIGHT, STAGE, and REARED were dropped from the ASCII version of the file. These were either virtually always empty or redundant with other data. They are still available in the DBF file, or from me by request. Second, all empty or null fields have been filled with an "X" to represent missing data. Any analytical processing of the data should properly identify the "X" as missing data.

For quantitative methods (SMPMETH = PIB or SUR) the actual count of individuals of that taxon in the sample is given in both the TXT and DBF files; so in these records a "1" represents only one individual in the sample. Most of the "counts" for the other sampling method types are null (SMPMETH = QUD, AER or RED); the mere existence of the record in the file indicates the presence of a specific taxon in that sample.

Other fields in the dataset were empty and are now filled with "X" -- reflecting the absence of corresponding data for that record. FUNCGRP is not available for some taxa, so its analytical value at all is highly questionable. A single "X" has been placed in each such field that is truly null.

The AQINS.TXT file is sorted (ascending) by the following fields, in this order: SMPDATE, SITE, ASMPL (which follows SMPMETH), and TAXA. This sort gives a "taxa-by-sample#-by -site-by-date" which was useful for checking items in the file. The data occupy the first 45 characters in the text file, including double spaces between actual data columns. A short description of data field identity, column location, maximum length, and some notes

on the data are given below. The range of "Columns" occupied for each field should be consulted for importing the data using a fixed column format into other programs.

Description of Fields (variables) in the AQINS.TXT file

-----S.Tessler 7/93 SNP

#	Name	Columns	MaxLngth	Note
1.	SITE	1-5	5	alpha-numeric LTEM site code
2.	QUARTER	8	1	range is 1 to 4
3.	SMPDATE	11-18	8	date format is numeric YYYYMMDD
4.	ASMPL	1	1	range is 1 to 6 per SMPMETH
5.	SMPMETH	24-26	3	PIB, SUR, QUD, AER, RED
6.	TAXA	29-35	7	codes refer to TAXADICT.DBF
7.	INSCNT	38-41	4	for PIB & SUR "1" is real count
8.	FUNCGRP	44-45	2	see LTEMs manual; some null (X)

<sup>i</sup>Content Standards for Digital Geospatial Metatdata Workbook, version 1.0, Federal Geographic Data Committee, March 24, 1995