United States Environmental Protection Agency

Office of Research and Development

National Human Exposure Assessment Survey (NHEXAS)

Arizona Study

Quality Systems and Implementation Plan for Human Exposure Assessment

The University of Arizona
Tucson, Arizona 85721

Cooperative Agreement CR 821560

Standard Operating Procedure **SOP-UA-D-16.0**

Title: First Stage of Cleaning Electronic Data (Hand Entry)

Source: The University of Arizona

U.S. Environmental Protection Agency
Office of Research and Development
Human Exposure & Atmospheric Sciences Division
Human Exposure Research Branch

Notice: *The U.S. Environmental Protection Agency (EPA), through its Office of Research and Development (ORD), partially funded and collaborated in the research described here. This protocol is part of the Quality Systems Implementation Plan (QSIP) that was reviewed by the EPA and approved for use in this demonstration/scoping study. Mention of trade names or commercial products does not constitute endorsement or recommendation by EPA for use.*

First Stage Cleaning of Electronic Data

1.0 Purpose and Applicability

The purpose of this procedure is to provide a standard method for the "first stage" of cleaning data. The first cleaning stage takes place after data verification and before master database appendage.

This procedure applies to (1) post-keypunch data collected by the NHEXAS Arizona staff in hard copy form, and (2) uncleaned data corresponding to the hard copies in electronic (i.e., database) form.

2.0 Definitions

2.1 AHSC SITE: Arizona Health Sciences Center, located at 1501 N. Campbell Avenue; University of Arizona; Tucson, AZ 85724. The Respiratory Sciences Center is based at this site.

2.2 ASCII DATA FILE: An ASCII file which holds data to be checked by a specific dictionary. This data will be stored one record per line with each variable starting and ending at predefined locations (determined by the lengths of the variables before them and their own lengths). If the dictionary checks multiple tables (as stored in the database), then each record is broken up into tables and each table is stored one per line (rather than one record per line). Thus, a record consisting of *n* tables will be represented in *n* consecutive lines.

2.3 BACKUP: (v.) The process of creating a duplicate of a file, directory, or drive to protect against data loss during a hardware or software failure. (n.) The duplicate copy created during this process.

2.4 BERNOULLI BOX: A peripheral mass storage device with removable data media (i.e., Bernoulli).

2.5 CODE: Classified under this word are the following definitions: CODE, COMMENT; CODE, ERROR; CODE, GLOBAL; CODE, VERIFICATION.

2.5.1 CODE, COMMENT: A numerical code designating the validity of a field sample or questionnaire response. Each code indicates a status of either "valid," "suspect," or "void."

2.5.2 CODE, ERROR: A letter code indicating why a correction was made to a physical data form.

2.5.3 CODE, GLOBAL: A set of standard codes used in data within the Respiratory Sciences Center designating the status of a data field in three cases: datum refused, datum non-applicable, and datum missing.

2.5.4 CODE, VERIFICATION: Used specifically by the data cleaning application *Data.exe*, this is a numerical code assigned to the variable "VERIF" in a data record. The verification code designates the level of data processing completed on a data record, from data entry to range checking.

2.6 DATA: Classified under this word are the following: DATA CLEANING; DATA, ELECTRONIC; DATA, ENTERED; DATA, LOGIC CHECKED; DATA, MISSING; DATA, PHYSICAL; DATA PROCESSING BATCH; DATA PROCESSING ERROR; DATA PROCESSING LEVEL; DATA RECORD; DATA, RANGE CHECKED; DATA, UNCLEANED; DATA VALIDATION; DATA, VERIFIED.

2.6.1 DATA, ELECTRONIC: Data stored on some type of magnetic or optical medium (for example: floppy disk, hard disk, Bernoulli, tape).

2.6.2 DATA, ENTERED: Electronic data entered for the first time into a computer database. Entered data are the product of "data entry."

2.6.3 DATA, LOGIC CHECKED: Data records that were checked for and cleared of all apparent logical errors. Logic checked data are the product of "logic checking" or "running dictionary" (see below).

2.6.4 DATA, MISSING: A datum or data that was applicable to a sample or question, but was not recorded on the physical data form at the time of initial observation; or one or more data records that were applicable to a data processing batch, but were not appended to the working or master database at the time of initial data appendage.

2.6.5 DATA, PHYSICAL: A datum or data written on a physical data form.

2.6.6 DATA, RANGE CHECKED: A data record where the value of each variable in the record was compared against a preestablished valid range. If any values are out of range and valid, then the range parameters are re-defined. If any values are out of range and invalid, then they are corrected.

2.6.7 DATA, UNCLEANED: Entered and verified electronic data that are not yet cleaned (See DATA, ENTERED and DATA, VERIFIED).

2.6.8 DATA, VERIFIED: Electronic data re-entered into the same table and database into which it was originally entered, and programatically compared against the original entered data. Verified data are the product of "data verification."

2.6.9 DATA CLEANING: The process of locating and correcting data processing errors (see DATA PROCESSING ERROR below). They can be individual level errors in the electronic and physical data, or they can be system level errors in the data collection, packaging, coding, entry, and cleaning procedures themselves. This process is also referred to as "data validation."

2.6.10 DATA PROCESSING BATCH (DP BATCH): A collection of household packets or physical data forms reviewed for quality assurance and ready for data entry. Each DP batch receives a unique numeric or alphanumeric code that is written on all forms in the DP batch and is entered into the database corresponding to that form.

2.6.11 DATA PROCESSING ERROR: An error occurring at any level of data processing (see DATA PROCESSING LEVEL below). It is a procedural mistake, such as a duplicate data record, a typographical error, a logical error, or missing information.

2.6.12 DATA PROCESSING LEVEL: This refers to the various levels of data processing: (1) data collection in the field, (2) assembly of physical data, (3) coding of physical data, (4) data entry, (5) data verification, (6) data cleaning or validation, (7) range checking, (8) logic checks, and (9) data analysis.

2.6.13 DATA RECORD: In the context of this SOP, this is a row of electronic data in a database.

2.6.14 DATA VALIDATION: See DATA CLEANING.

2.7 DATABASE: Classified under this word are the following: DATABASE, EMPTY; DATABASE, MASTER; DATABASE, WORKING.

2.7.1 DATABASE, EMPTY: An R:BASE database structure that contains no data records except for parameter rows; or a non-R:BASE database structure that contains no data records.

2.7.2 DATABASE, MASTER: Accumulative database generated from the appendage of newly cleaned data processing batches. *Copies* of master databases are used in analyses and any data corrections made to copies are also made to the master databases. Thus, a master database is the most complete and accurate database of its kind.

2.7.3 DATABASE, WORKING: A database earmarked for or in the process of cleaning that contains one or more data processing batches. When cleaned, it will be appended to the master database.

2.8 DICTIONARY: A program that evaluates the electronic data records for a specific physical form and performs basic range-checking and logic-correctness functions. The electronic data records are stored in an ASCII representation, to be referred to as the ASCII DATA FILE (see above). (See SOP# UA-D-4.0 for more information about dictionaries.)

2.9 DISKETTE (OR FLOPPY DISK): A small data storage medium used for storing or transferring small amounts of data. Diskettes used in this project are usually high density double-sided disks in two sizes: 3.5 inch (1.44MB) and 5.25 inch (1.2MB).

2.10 FIELD: An area on a data entry form (i.e., screen) where a datum from a physical form is entered (see FORM, DATA ENTRY below).

2.11 FILE PROTECTION: A UNIX security system that limits who has access to a file either within a user group or as an individual user.

2.12 FORM: Classified under this word are the following definitions: FORM, DATA ENTRY; FORM, PHYSICAL.

2.12.1 FORM, DATA ENTRY = A computer screen representation of a physical data form. The data on the physical form is entered into the computer database via the data entry form.

2.12.2 FORM, PHYSICAL = The paper or "hard copy" version of a data form. This is also referred to as a "physical data form."

2.13 HOUSEHOLD IDENTIFICATION (HHID): A four-digit integer that uniquely identifies a study residence.

2.14 HOUSEHOLD IDENTIFICATION (HHID), SPECIAL: A four-digit integer assigned to a special site or household lacking respondents. Special HHIDs are not to be confused with the global codes for HHID (see CODE, GLOBAL above and SOP# UA-D-5.0).

2.15 HRP SITE: The Health Related Professions building, located at 1435 North Fremont Avenue; Tucson, AZ 85719. This is an annex of the Respiratory Sciences Center and the primary site of NHEXAS Arizona.

2.16 IPOMEA: A SparcStation housing working databases at the HRP site.

2.17 KEY VARIABLE(S): One or more variables in a data record whose value or combined values make a data record unique from the others in the same table or file.

2.18 KEYPUNCH: The primary area in which data entry and data verification of NHEXAS Arizona field data takes place. It is located in the Respiratory Sciences Center, Room 2329; Arizona Health Sciences Center (AHSC); 1501 N. Campbell Avenue; University of Arizona; Tucson, AZ 85724.

2.19 LAN: Abbreviation for Local Area Network. This is any physical network technology that operates at high speed over short distances.

2.20 LONICERA: A SparcStation housing working databases at the HRP site.

2.21 NHEXAS Arizona: Acronym for National Human Exposure Assessment Survey, a research project conducted in Arizona by the University of Arizona/Battelle/Illinois Institute of Technology consortium.

2.22 NODE: A computer that is attached to a network; also called a host.

2.23 NULL: A value in a data field that contains the database software symbol for system missing. This means that no value exists for this field. A null value is not necessarily a zero value.

2.24 PACK: Compression of a file to eliminate empty data records so it takes up less computer disk space. This is done within an R:BASE database with the command "pack"; or to one or more files using a utility program such as pkzip.

2.25 PACKET: A sturdy, envelope-like container that can be fully closed and is large enough to hold the physical data form(s) generated by a study household, laboratory, research site, or data processing batch. One type of packet is used for one type of physical data forms (eg., manila envelopes would be used for all lab forms processed at the HRP site).

Packets are either color coded, labeled according to their contents, or both.

2.25.1 PACKET, DATA: A small unit of data transferred electronically from one machine (i.e., computer) to another (i.e., computer, printer). It usually contains only a few bytes. It is not to be confused with a household packet, a lab packet, or a site packet.

2.25.2 PACKET, HOUSEHOLD: A packet containing the physical data forms for a study household.

2.25.3 PACKET, LAB: A packet containing the physical data forms generated during laboratory evaluation of field samples.

2.25.4 PACKET, SITE: A packet containing the physical data forms for research sites.

2.26 PARAMETER ROWS: The five data records in an R:BASE table that define the following for each variable contained therein: (1) the upper range limit, (2) the lower range limit, (3) the global code for refusal, (4) the global code for non-applicable, and (5) the global code for missing.

2.27 PC: Abbreviation for Personal Computer. This is a microcomputer based on the Intel 8088/8086 instruction set. The HRP site has eleven operating PC machines. Their node names are: Cereus ('386/33MHz), Datura ('286), Cycad ('286), Ginko ((8088 with a '286 accelerator), Puccinia ('286), Cieba ('286), Acacia ('386/25), Salix (386 laptop), Ficus ('286, no keyboard, no monitor -- Ficus is used as a SLIP router only), and Ephedra ('386/25). There is also an 8086 based laptop with no node name (unconnected to the LAN).

2.28 QXV: Abbreviation for questionnaire version. The QXV is a code assigned to NHEXAS Arizona physical forms that designates its iteration.

2.29 RESEARCH STAGE: A sampling stage of NHEXAS Arizona.

2.30 RESPONDENT: A subject in the NHEXAS Arizona study population.

2.31 RUNNING A DICTIONARY: The act of using a Statistical Package for the Social Sciences (SPSS) version of the dictionary (although not necessarily a working version) on the ASCII DATA FILE with the purpose of cleaning it.

2.32 SCORAZ: The SparcStation at the AHSC site that houses NHEXAS Arizona databases having one or more of the following statuses: (1) empty databases ready for data entry, (2) databases containing entered data, and (3) databases containing verified data. The status of any given database depends on the status of the current data processing batch.

2.33 SPARCSTATION: A small computer developed by Sun Microsystems and based on a RISC (Reduced Instruction Set Computer) Processor. The Sun SparcStation is a fast, multi-tasking, multi-user platform that runs the SunOS version of the Unix operating system. The node names of the Sun SparcStations at the HRP site are ipomea and lonicera.

2.34 SUBJECT: A respondent in the NHEXAS Arizona study population.

3.0 References

3.1 *Bernoulli Box: User's Manual.* IOMEGA Corp. 1985

3.2 *Microsoft MS-DOS Operating Systems V5.0 User's Guide and Reference.* Microsoft Corp. 1984.

3.3 *R:BASE for DOS: Building Applications Command Dictionary* (First Edition, Version 2.0). December, 1987. Redmond: Microrim, Inc.

3.4 *R:BASE for DOS: User's Manual* (First Edition, Version 2.0). December, 1987. Redmond: Microrim, Inc.

3.5 *R:BASE for DOS: Utilities Manual* (First Edition, Version 2.0). December, 1987. Redmond: Microrim, Inc.

3.6 *R:BASE for DOS: Command Summary* (First Edition, Version 2.0). December, 1987. Redmond: Microrim, Inc.

3.7 *R:BASE for DOS: Developer's Express Guide* (First Edition, Version 2.0). December, 1987. Redmond: Microrim, Inc.

3.8 *R:BASE for DOS: Learning Guide* (First Edition, Version 2.0). December, 1987. Redmond: Microrim, Inc.

3.9 *R:BASE Program Interface User's Manual* (Version 2.0.) April, 1987.

3.10 *Readme.doc* (electronic file document). Pkzip Version 2.04g. February 1, 1993. Brown Deer: PKWARE, Inc.

3.11 *Sharewar.doc* (electronic file document). Pkzip Version 2.04g. February 1, 1993. Brown Deer: PKWARE, Inc.

4.0 Discussion

The goal of cleaning data is to make the data as accurate and as complete as possible, leaving no fields empty or uncoded. Any data fields left uncorrected or uncoded will eventually have to be corrected.

Although data are cleaned at all levels of data processing, the first stage of cleaning electronic data is the most important for four reasons. First, data corrections are most easily made while the data is still in the database format, as opposed to a flat ASCII or SPSS format. Second, it is the most efficient time to locate and resolve problems because there are often greater time constraints at the analysis stage. Third, it is the main source of confidence in the master database since the master is built from data cleaned at the first stage. Fourth, data cleaned well at the first stage of cleaning will limit data problems encountered during statistical analysis.

4.1 Principles of the Method

4.1.1 Backup of Electronic Data Off of LAN

A backup of the uncleaned and cleaned electronic data for each data processing batch is made off of the LAN system (i.e., on a floppy diskette or a Bernoulli). This is a security measure against LAN system failure.

4.1.2 Consistency

Data are cleaned in a consistent manner throughout a research stage of NHEXAS Arizona.

4.1.3 Independent Validation

An intersubjective approach to data validation is employed. This helps to control any bias in data cleaning that would be produced if one person were to work with the same data at all levels of data processing. For example, if one person were to enter a batch of missing data, then another would verify it.

5.0 Responsibilities

5.1 The Project Data Coordinator is responsible for the following:
(a) Supervising the cleaning of all working databases used by NHEXAS Arizona;
(b) Maintaining consistency in coding and cleaning procedures;
(c) Providing and/or approving updates to coding or cleaning procedures;
(d) Consulting with the On-Site Principle Investigator, the Project Data Analyst, and/or the Project Field Coordinator in determining changes in coding and cleaning procedures, on an "as needed" basis;
(e) Creating and updating the subdirectories on ipomea or lonicera in which working databases are cleaned;
(f) Copying the working databases from Scoraz to the appropriate subdirectories on ipomea or lonicera for cleaning;
(g) Changing file protections for accessibility by the "HandE" user group on all cleaning-related files and subdirectories on ipomea or lonicera;
(h) Making and retaining backups of all uncleaned working databases retrieved from keypunch;
(i) Making a retaining backups of all cleaned working databases, whether from keypunch or elsewhere;
(j) Approving any corrections made to working databases and/or the physical form associated with it.

5.2 The Project Data Analyst/Data Manager is responsible for the following:
(a) Changing file protections to cleaning-related files when necessary;
(b) Resolving problems with computer hardware, software, and the computer network system when possible (see SOP #UA-D-1.0 for an elaboration of the Project Data Manager's responsibilities).

5.3 The Student Data Assistant is responsible to the Project Data Coordinator, and is responsible for the following:

(a) Keeping the physical data forms in a safe place while the data processing batch is earmarked for, or in the process of being cleaned, until the physical forms are filed;

(b) Documenting data cleaning steps on the appropriate batch tracking form;

(c) Documenting all corrections made to the physical and/or electronic data on the appropriate data correction log form (see SOP# UA-D-25.0 for database correction procedures);

(d) Printing, signing, and dating any database correction files generated from range checked data;

(e) Submitting any database correction documentation to the Project Data Coordinator;

(f) Performing the first stage of data cleaning on working databases according to this general SOP and to the specific SOP for the data being cleaned.

(g) Making and retaining daily backups of current projects off of the LAN (i.e., on floppy disk or bernoulli labeled with name of user and clear indication that current projects are contained therein);

(h) Investigating data processing errors at other levels of data cleaning when necessary.

(i) Performing other tasks under other SOPs as needed.

5.4 The Project Field Coordinator is responsible for providing information regarding the standard operating procedures for the preparation of the household packets and the collection of field data.

5.5 The Student Field Technician is responsible to the Project Field Coordinator and also for providing to the Project Data Coordinator specific information about the physical data that he or she has helped to collect.

6.0 Materials and Reagents

6.1 Materials

6.1.1 Data Cleaning Documentation Forms and Media (See Attached Documents)

(a) **"Data Validation Records" Form (Figure 1):** This form serves as a record for the location and cleaning status of each working database. It contains the DP batch number, a description of the DP batch, the name and location(s) of the working database, a description of any entered or verified data, the initials of the person who has cleaned the data, and the dates that each cleaning step was completed. This form is referred to as the "batch tracking form" in this SOP.

(b) **"Data Entry and Validation Records II" Notebook:** This notebook houses the batch tracking forms and the batch append forms. It is referred to as the "DEVR-II notebook" in this SOP.

(c) **"Working Database Change Log Form" Form (Figure 2):** This form is used to document corrections made to the working database(s). It includes the database name, name(s) of file(s) or table(s), key variables, name of variable corrected, original value, new corrected value,reason for the correction, whether a correction was made to the physical form, initials of the person who made the correction, date of correction, and initials of the project data coordinator. This form is referred to as the "data correction log form" in this SOP.

(d) **"Master Database Appendage Record" Form (Figure 3):** This form contains a record of the cleaned working databases ready for appendage to the master databases, the initials of the person who appended them, and the date appended. It is referred to as the "batch append form" in this SOP.

(e) **"Cleaning Check Sheet" Form (Figure 4):** This is a detailed list of the steps involved in cleaning a data processing batch of household packets. The Student Data Assistant puts a checkmark next to a step after he or she completes it.

(f) **"Cleaning Problems Sheet" (Figure 5):** This is a form used to keep track of problems that are pending resolution by either the Project Data Coordinator or by other sources. It serves as both a data cleaning aid and as a written record of how problems were resolved. The latter will ensure the consistent treatment of problems throughout a research stage.

(g) **"DP Batch Archives I" Bernoulli:** This bernoulli stores compressed copies of cleaned, working databases for each data processing batch. It is referred to as the "DPBA-I bernoulli" in this SOP.

(h) **"Cleaning Folder(s)"** labeled with the title of the physical form being cleaned

6.1.2 Coding and Cleaning Documents

(a) Coding instructions for the physical form to be cleaned
(b) Comment Codes
(c) Error Codes
(d) SOP# UA-D-5.0: Global Coding Used by NHEXAS Arizona
(e) Specific data cleaning SOP for the data being cleaned

6.1.3 Hardware: It is necessary to work on the LAN. This requires use of a PC that is connected to the Respiratory Sciences Center LAN where SparcStations ipomea or lonicera are network nodes.

6.1.4 Supplies
(a) Floppy diskettes (5.25" or 3.5" high density) or bernoulli for off-LAN backups
(b) "Post-it" Notes
(c) Stapler and staples
(d) Paper clips or clamps
(e) Binder clips
(f) Cellophane tape
(g) Rubber bands
(h) Purple pen

6.1.5 Software
(a) R:BASE for DOS, Version 2.11
(b) Pkzip Shareware, Version 2.04g

6.1.6 Cleaning Applications: There are two applications used for all uncleaned, R:BASE databases:

(a) *Data.exe* (date = 7/11/90; 5:37 pm; 124,782 bytes): This executable C program works with all R:BASE databases. It is used for data entry, data verification, range checking, and data correction documentation.

(b) *Dup.exe* (date = 7/14/89; 1:35 am; 79,820 bytes): This executable C program checks for duplicate key variables by table(s) and reports on them. It does not delete any data records. The output is sent to a file called *<database>.dup*.

(c) Other Form-Specific Applications: Consult the specific data cleaning SOP for the data being cleaned for any other relevant cleaning applications. See Appendix 1 for a complete listing of cleaning applications and the data for which they are used.

6.1.7 Data

(a) Entered and verified uncleaned electronic data for the data processing batch(es) to be cleaned

(b) Physical data forms for the data processing batch(es) to be cleaned

6.2 Reagents: Not Applicable

7.0 Procedure

7.1 Preparations

7.1.1 Site Selection Criteria: Not Applicable

7.1.2 Record today's date after "Start Date:" on the batch tracking form in the DEVR-II notebook.

7.1.3 Segregate and Visually Review the Physical Forms

(a) Remove the physical forms from the household packets and sort them into piles by form and by HHID in ascending order.

(b) While sorting, visually scan them for obvious problems such as missing key variables. Note any problems in writing.

(c) Secure each pile of forms with a binder clip, clamp, or giant paper clip -- whichever is most effective. This reduces the potential for physical form loss.

(d) Place each pile in the appropriate "cleaning folder."

(e) Place the "cleaning folders" in standing files within your primary work area.

7.1.4 Post-Keypunch Coding

Consult the Project Data Coordinator and the specific data cleaning SOP for the physical form being cleaned for any post-keypunch coding. Some forms require comment codes; others may have uncoded fields that were pending further research by the Project Data Coordinator at the time of initial coding.

7.2 Data Cleaning

7.2.1 Use the specific data cleaning SOP for the data to be cleaned as a supplement to this general cleaning SOP.

7.2.2 Log into the LAN with your username and password. If you lack a user account, then obtain one from the Project Data Manager before proceeding.

7.2.3 If a "Keypunch Problem Sheet" (Figure 6) accompanied the data processing batch, then review it for the following problems:

(a) Physical forms not entered
(b) Physical forms not verified
(c) Typographical errors of key variable(s)
(d) Missing key variable(s)

7.2.4 If any of the above problems listed in 7.2.3 above are present on the "Keypunch Problem Sheet," then correct them before proceeding. Do it in the following way for the above problems, respectively:

(a) A first Student Data Assistant enters all physical forms that were not entered by using *Data.exe* on the working database. He or she logs his or her initials and current date on the "Data Validation Records" form after "Ent by:".
(b) A second Student Data Assistant verifies all physical forms that were not verified by using *Data.exe* on the working database. She or he logs her or his initials and current date on the "Data Validation Records" form after "Ver by:".
(c) Correct any typographical errors for key variables in the electronic data.
(d) Correct any missing key variables. Missing keys will jeopardize range checking.

7.2.5 Go to the appropriate directory on ipomea or lonicera that houses the working database in process. Its location is on the "Data Validation Records" form. You may access it through PC-NFS on the m: drive or through telnet.

The standard subdirectory structure of m:\workstor is as follows: Second level subdirectories named after each working database branch off of m:\workstor. For each second level subdirectory, one or more third level subdirectories are defined which have a two-part name. The first part is the name or abbreviation of the data entry form contained in the database. The second part is the data processing batch(es) contained in the working database. These two parts are separated by a hyphen.

For example, the working database for DP batch number one of the "Demographic Questionnaire" would be stored in the directory m:\workstor\demog\demog-1, where *DEMOG* is the database name, *DEMOG* is the data entry form name, and one is the DP batch number. Thus, in general, the working database is stored in

```
m:\workstor\ <database> \ <data entry form or abbreviation> - <DP batch number(s)>.
```

7.2.6 Pre-Range Check Frequencies
(a) Tally the variable "DPBATCH" where the variable "VERIF" is greater

than -1, which are all data records except the parameter rows.

(b) Compare the results with the number written on the keypunch batch tracking form. If the DP batch number(s) does not match, then try to resolve it. If necessary, seek help from the Project Data Coordinator.

(c) Tally the variable "VERIF" (first pass) where the variable "VERIF" is greater than -1, which are all data records except the parameter rows. Verify any data records with a verification code of null (-0-) using *Data.exe*.

(d) Establish that all non-parameter data records have a verification code equal to three before proceeding.

7.2.7 Check for Duplicates

(a) Run the Program DUP.EXE.

1. List ALL of the key variables in the command.
2. Print the output file <database>.dup if duplicate key variables have been found.
3. If the a data record's keys are listed twice, then there are three data records with the same key variables.

(b) Look at the alleged duplicate data records and compare their values to each other and to the physical form with the same key variables. A "false duplicate" can occur when a key variable has been misentered or the data on the physical form were incorrect. Thus, check all values carefully.

(c) Note any discrepancies between or among the data records and keep a record of which row contains the more accurate values.

(d) If no discrepancies are found between or among the duplicate data records, then delete all of them except for one row. If one or more discrepancies are found, then make sure that one row is completely correct and delete the other(s).

(e) Pack (or compress) the database if any data records were deleted.

7.2.8 Run Data.exe for range checks. Any value in a data field that is out of range will appear on the screen in reverse video. When this occurs, the user is given seven options for resolving the problem. The following is a guide to using these options.

(a). If value is out of range and an illogical response:

1). Request change; press the F5 key.

(b). If null value:

1). If the question is non-applicable (i.e., if the response to an earlier question instructed the respondent to skip the question; or it is non-applicable according to the field SOP), enter non-applicable; press F3 key.

2). If the form or question is refused, press the F5 key to document the correction.

3). If the datum can be gotten from some other source/form, request change; press F5 the key.

4). If the datum is missing, press the F2 key.

5). If the data record contains mostly or all null values and is a duplicate record, then it was probably entered by mistake. Record the key variable(s) on paper and press the F4 key throughout the record. This record will likely be deleted manually after range checking.

(c). If the value is out of range but is a logical response:
   1). Use the "keep" option; press the F4 key.
   2). Make a recommendation to change the parameter to the Project Data Coordinator.

7.2.9 Fill out the "Cleaning Problems Sheet" for all data processing errors that could not be resolved during range checks. Errors that are resolved after range checking are the following:

(a) Data Records Containing Mostly or All Nulls: A data record containing mostly or all null values is usually an accidental entry made by an inexperienced student data entry assistant. If it contains no key variable(s), it is invalid. If it does contain key variable(s) but corresponds to no existing physical data, then it is also invalid. Delete invalid records such as these from the database manually.

(b) Missing or Skipped Data Records: A missing or skipped data record needs to be entered and verified separately using Data.exe.

(c) Miscoded or "Botched" Data: Miscoded or "botched" data can often be seen on the screen. If data appear odd, then make a note of it and look at them manually.

7.2.10 Post-Range Check Frequencies

(a) Tally the variable "VERIF" (follow-up check) where the variable "VERIF" is greater than -1 (all data records except for the parameter rows).
   1. If one or more non-parameter rows is equal to three:
      a). Locate the problem and resolve it.
      b). Range check the data record(s).

(b) Establish that the variable "VERIF" equals four in all non-parameter data records before proceeding.

7.2.11 Initial and date the batch tracking form in the DEVR-II notebook after "RNG BY:" when finished range checking.

7.2.12 Pack (or compress) the database if any more data records were deleted.

7.2.13 Run dictionary. See SOP# UA-D-4.0 for the procedure.

7.2.14 Archive Cleaned DP Batch

(a) Make a compressed backup of the cleaned DP batch on the DPBA-I bernoulli using Pkzip.

(b) Initial and date the batch tracking form in the DEVR-II notebook after "ZIP BY:" when finished zipping.

7.2.15 Final Documentation

(a) Add the name(s) of the cleaned electronic file(s) and their location(s) to the append form.

(b) Turn in any data correction log forms to the Project Data Coordinator.

(c) Print <database>.cor file(s) if applicable.

(d) Initial and date the batch tracking form in the DEVR-II notebook after "COR BY:" if any corrections were made to the database.

(e) Inform the Project Data Coordinator that the DP batch is cleaned.

7.3 Calculations

See the specific cleaning SOP for the calculation procedure(s) if applicable.

7.4 Quality Control

7.4.1 Tolerance Limits

In order to test the accuracy of the validation procedures outlined in this SOP, a random comparison is made between the electronic data and the physical data for each cleaned data processing batch.

Ten percent of the data records are randomly selected and compared to the physical forms with the same key variable(s). If the error rate is greater than five percent, then a fifty percent random check is done. If the error rate exceeds five percent again, then the data are completely re-entered and the first stage of data cleaning begins anew (see UA-D-26.0).

7.4.2 Detection Limits

Any physical or electronic datum changed incorrectly and documented on the data change log form is detectable. In addition, ten percent of electronic data entered or cleaned incorrectly are detectable via the electronic data QA check.

7.4.3 Corrective Actions

Any physical or electronic data discovered to be incorrect will be corrected by either a Student Data Assistant or the Project Data Coordinator.

8.0 Records

8.1 Location of Data Processed by This Procedure

8.1.1 Electronic Data: The working databases are located in the workstor subdirectory of ipomea or lonicera until they are appended to the master databases. (See section 7.3.3 of this SOP for details about the workstor subdirectory.)

Also, as a security measure, compressed backups of the cleaned working databases are stored off of the LAN system on the DPBA-I Bernoulli.

8.1.2 Physical Data Forms: Physical Data forms are removed from packets for cleaning and are returned to their packets after cleaning. Complete packets are filed in cabinets by HHID and are locked in the Project Data Coordinator's office at the HRP site.

8.2 Record Forms (Attached)

Figure 1: "Data Validation Records" (example of batch tracking form)
Figure 2: "Working Database Change Log Form" (data correction form)
Figure 3: "Files to Append to the Master Database" (batch append form)
Figure 4: "Cleaning Check Sheet"
Figure 5: "Cleaning Problems Sheet"
Figure 6: "Keypunch Problem Sheet"
Appendix 1: Specific Applications for Special Subroutines

8.3 Location of Forms

Data cleaning and documentation forms are filed in notebooks and stored in the Project Data Coordinator's office or other data staff office at the HRP site.

Figure 1: "Data Validation Records" (example)

DATA VALIDATION RECORDS: HOUSEHOLD PACKETS

DP Batch#:

Form ID: UA-D-16.0-1.0

DATABASE DESCRIPTION AND LOCATION(S)	OTHER INFO & DATA TO ENTER/VERIFY	DATA CLEANING STEPS
Database (Rbase) BCHEM Table(s) BCHEM Form BCHEM Location(s) 1) m:\WORKSTOR\BCHEM\BCHEM-_____ 2) ______________ 3) ______________	[ ] discontinued [ ] enter/verify (describe):	**Start Date:**___/___/___ ENT BY_____ON___/___/___ VER BY_____ON___/___/___ RNG BY_____ON___/___/___ CHG BY_____ON___/___/___ COR BY_____ON___/___/___ ZIP BY_____ON___/___/___ APP BY_____ON___/___/___
Database (Rbase) BURKARD Table(s) INDBURK Form INDBURK Location(s) 1) m:\WORKSTOR\BURKARD\INDB-___ 2) ______________ 3) ______________	[ ] discontinued [ ] enter/verify (describe):	**Start Date:**___/___/___ ENT BY_____ON___/___/___ VER BY_____ON___/___/___ RNG BY_____ON___/___/___ CHG BY_____ON___/___/___ COR BY_____ON___/___/___ ZIP BY_____ON___/___/___ APP BY_____ON___/___/___
Database (Rbase) CHECK4 Table(s) PURMON Form PURMON Location(s) 1) m:\WORKSTOR\CHECK4\PUR-_____ 2) ______________ 3) ______________	[ ] discontinued [ ] enter/verify (describe):	**Start Date:**___/___/___ ENT BY_____ON___/___/___ VER BY_____ON___/___/___ RNG BY_____ON___/___/___ CHG BY_____ON___/___/___ COR BY_____ON___/___/___ ZIP BY_____ON___/___/___ APP BY_____ON___/___/___
Database (Rbase) DUSTMIT Table(s) FLDMITE1 FLDMITE2 Form DUSTMITE Location(s) 1) m:\WORKSTOR\DUSTMIT\MITE-___ 2) ______________ 3) ______________	[ ] discontinued [ ] enter/verify (describe):	**Start Date:**___/___/___ ENT BY_____ON___/___/___ VER BY_____ON___/___/___ RNG BY_____ON___/___/___ CHG BY_____ON___/___/___ COR BY_____ON___/___/___ ZIP BY_____ON___/___/___ APP BY_____ON___/___/___

Start Date = date cleaning process began
ENT = entered data
VER = verified data
RNG = range checked data

CHG = made all additional corrections to database and physical data forms
COR = printed *.cor file(s) and/or turned in change forms for approval
ZIP = made *.zip of cleaned database on "Cleaned Database Zips" bernoulli
APP = logged name of cleaned database on "Files to Append" form

Figure 2: "Working Database Change Log Form"

DP Batch(es):__________

DATABASE: **WORKING DATABASE CHANGE LOG FORM**

KEY1	KEY2	File(s) or Table Name	SUGGESTED CHANGE:			Reason for Change	STAFF OK	CHANGED IN DATABASE:		
			Var.	From	To			By	Date	SD*

* SD stands for "Source Document." Check this column if you have made a change in purple ink to the physical data sheet.

Figure 3: "Files to Append to the Master Database"

CLEANED FILES TO APPEND TO THE MASTER DATABASE

FORM NAME: ______________________________

DATABASE/FILE NAME(S): ______________________________

DP Batch	Location of Database	Comments	Appended:	
			Init.	Date
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			
	m:\workstor\________________			

Figure 4: "Cleaning Check Sheet"

Cleaning Check Sheet

Form:_________________________ Batch:____ Packet Color:_____

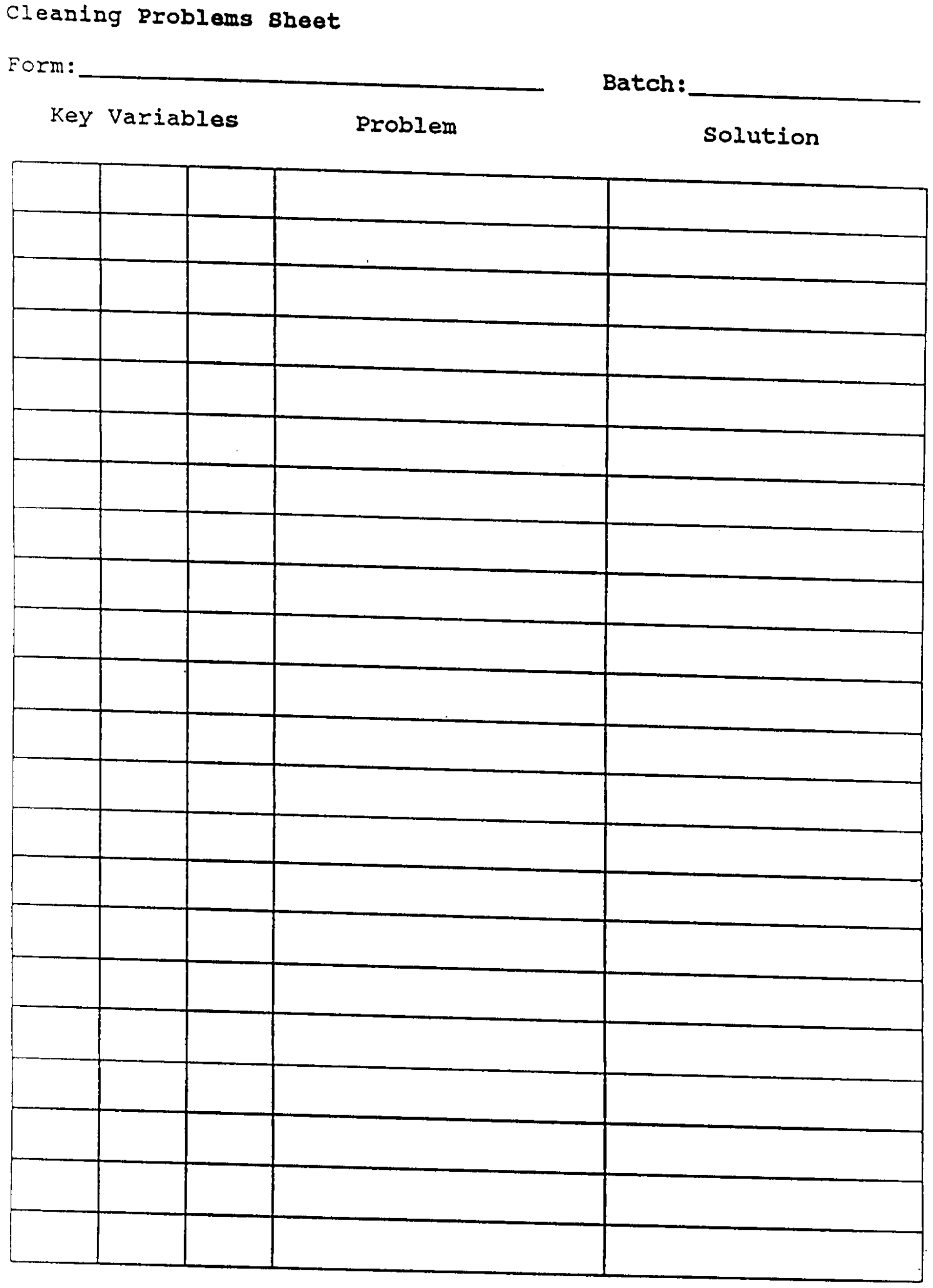
Location:____

Database:_______________________ Table:___________ Form:__________

Keys:____

___ Entered "Date Started" in DP Log Book
___ Tally VERIF and DPBATCH --> check for problems
___ Do: Dup [DATABASE] 1 [TABLE] [KEY VARIABLES]
___ Data from old forms transferred to new form
___ Stray sheets entered and logged
    Use: data enter [DATABASE] [FORM NAME]
___ Stray sheets verified and logged
    Use: data verif [DATABASE] [FORM NAME]
___ Everything range checked and logged
    Use: data range [DATABASE] [FORM NAME]
___ Miscellaneous questions answered
___ Changes made in database
___ Tally VERIF --> check for all "4"s
___ Pack database
___ .COR printed, initialed and put in LLF's Mailbox
___ Zipped and logged
___ Written on to be appended form and logged

Figure 5: "Cleaning Problems Sheet"

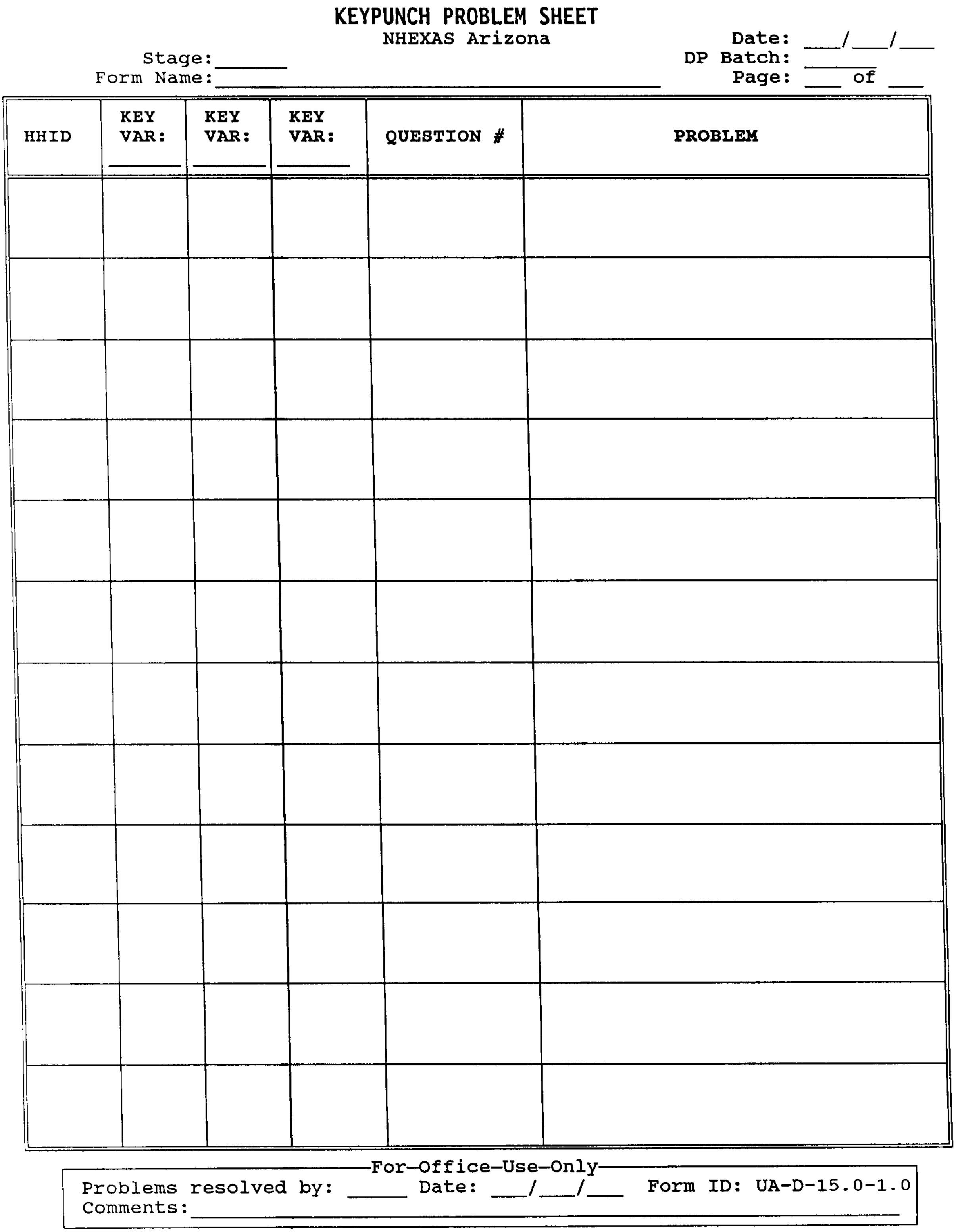


Cleaning **Problems Sheet**

Form:____________________________ **Batch:**______________

Key Variables			Problem	Solution

Figure 6: "Keypunch Problem Sheet"



KEYPUNCH PROBLEM SHEET

NHEXAS Arizona

Stage:_______

Form Name:_______________________________________

Date: ___/___/___

DP Batch: _______

Page: ___ of ___

HHID	KEY VAR: ______	KEY VAR: ______	KEY VAR: ______	QUESTION #	PROBLEM

For—Office—Use—Only

Problems resolved by: _____ Date: ___/___/___ Form ID: UA-D-15.0-1.0

Comments:____

Appendix 1: Specific Applications for Special Subroutines

DATA CLEANING APPLICATIONS FOR R:BASE DATABASES

DATA.C 07-11-90

This is the main program for data entry, verification, and validation. It is mostly used on data from research phases three and four. It can be used for data from research phase two, but is more cumbersome when there is more than one table and data entry form associated with a single physical form.
syntax: data <option> <database name> <form name> <scrolling>

<option> ::= "enter" | "verif" | "range"

enter = enter the data the first time
verif = enter data a second time, checking the second entry against the first
range = do range checking on verified rows of data

<database name> ::= any R:BASE database name

<formname> ::= the name of a form in R:BASE System V or R:BASE FOR DOS format

<scrolling> ::= "" | "noscroll"
noscroll = regions do not scroll

DATAMOVE.C 06-13-90

This application was previously entitled "move.c", and its built-in syntax still begins with the word "move". It was renamed to "datamove.c" because it conflicted with the DOS command "move".

This program is typically used to append validated data to a master R:BASE database. It moves data records (excluding any parameter rows) from one copy of a database to another in a different drive or subdirectory. It allows for checking to insure the original and target databases are compatible.

syntax:
datamove <database> <from | at> <"location"> to <"location"> checking

DUMMY.C 06-10-88

This application adds "dummy rows" of global codes to tables in a database. It fills each field in a data record with the global codes for refusal, non-applicable, or missing, with the exception of the key variable(s). The user needs to change by hand the values in the data record of any non-key field(s) that is (are) necessary for the identification and status of the record (eg., QXV, PKTCOLOR, DEDATE, VERIF, etc.).

syntax:

```
dummy <database> <# of tables> <table names...> <key names...>
```

DUP.C 07-14-89

This application looks for key variable duplicates in R:BASE databases and simply reports on them. It does not delete any data records.

syntax:

```
dup <database name> <#of tables> <table names...> <key names...>
```

FDATE.C 1-25-90

This program makes certain that a date is in this century.

syntax: fdate <database name>

FTEXT.C 1-24-90

This program is used to make certain that text fields are not filled with "garbage".

syntax: ftext <database name>

MATCH.C 5-04-88

This program compares record size for two different databases and reports any differences. It also compares copies of a database in different subdirectories. If the datamove command refuses to move data because the database definitions do not match in every respect, this can be used to discover the differences. As this has been made part of the Datamove program, the subroutine is rarely used anymore.

syntax: offset <database> <path> <path2>

<database1> ::= database to compare

<path1> ::= path for first copy of the database

<path2> ::= path for second copy of the database

GENERAL APPLICATIONS

FORM.C 02-13-89

This program prints sampler assignment sheets for any of three types of field samplers: (1) formaldehyde passive (FP); (2) nitrogen dioxide passive (NP); or (3) particulate matter (PM). It assigns sampler identification numbers to field blanks and batch blanks. It is interactive.

syntax: form