To all recipients of the CPP 'Gamma' version 7/28/2002. Jlawlor@jhsph.edu 703-897-8497

The two CDs represents the first time the disparate pieces of work to date have been assembled. This is very much unfinished/in progress, it was done with few resources, and it undoubtedly has many errors, large and small. Broadly, they contain:

1. The complete original public use data as ascii files and microfiche documentation as pdf files on one CD

2. SAS syntax to read in and reshape most of the data from the master data fileÆs punchcard format. These may be reused as is/rerun after adaptation to any host system, used as a guide for SAS or other input syntax, and we have created:

3. 5 large datasets with selected data from the master data file as SAS/WIN *.sas7bdat, SPSS portable *.por and STATA *.dta (the Stata datasets have been split - we recommend using SAS or SPSS if possible). SAS transport files .xpt are not provided - do users want them?

4. Edited original documentation and extra, draft, documentation.

5. The obstetric data has been programmed, but no system datasets created.

The recipients are asked, for this version

1. To comment if the overall product is what newcomers to the data might need to use it well.

2. Correct MAJOR errors, suggest what MUST be added for ease of use, and point out only the most IMPORTANT examples of lack of clarity or ambiguity.

The CDs and documentation have been created with a future hypertext/WWW version in mind. Therefore, no linearly written manual is provided. Think of links between modular, often redundant, explanations. Should this linear manual be written?

Again, is this what would YOU want to receive if you knew nothing of the cpp or its electronic format? Your suggestions will be implemented, resources permitting.

The final version will have much more on the CPP's background and history and bibliography.

STATA users are warned that there is considerable difficulty in either reading the data directly into Stata from ASCII or using the dbstata engine in SAS and DBMS/ENGINES. Furthermore, Stata reads a complete dataset into RAM, so a 150mb .dta file requires>150mb of RAM - usually 256mb.

FYI, some tasks in progress are: type variable value labels/all of hardcopy 1978 definition of codes (800 pages), bookmark the reformatted 75 fiche/Adobe files to a structure reflecting their content and the data now being off the mainframe, and scan clear copies of the forms - many are blurred on the microfiche. If you have any of this typed or scanned, and are willing to donate it, we will gladly accept it for inclusion.

If feedback is from someone else than yourself, please have them mention who they got the CD from, and a little on their history with the cpp, interest in it, would they consider themselves senior intermediate or beginning analysts, statisticians.or programmers? Anyone who gives feedback or donates syntax will be acknowledged in the final version. EXCEPT FOR EVALUATION PURPOSES, PLEASE DO NOT COPY IT OR DISTRIBUTE IT.

Most of you will not be happy with this modernization as it is a compromise of: computing at the limits of even modern pc's and software's capacities, clarity and complexity, severely limited by the scanty resources available.

But, its undisputable contribution is it contains the original public use mainframe data as 32 ASCII files for a PC, including the serological test results, and the cumbersome microfiche documentation as 75 much more manageable Adobe pdf files. There is, while not perfect, much more to guide and assist the analyst, statistical programmer or data manager.

Thanks (and ouch) in advance,

Reader: make full use of the GLOSSARY.pdf

Aim Supply information sufficient in range, depth, and brevity for a newcomer to the CPP to quickly, accurately, and ‘easily’ analyze the data but understand and appreciate its complexity.

Document the CPP data as found in the public domain at NARA in 2002.

Describe the rationale for, and document the technical aspects of, its manipulation into the system datasets supplied to aid the end user in tailoring the data to their specific needs; be they analyst, statistician, or programmer, seasoned CPP analyst or newcomer.

Audience: research analysts statistical programmers electronic data managers

Beginning, intermediate, and senior levels of each: including Fellows in the Department of Pediatrics, Johns Hopkins School of Medicine.

Few have substantial expertise in all three areas, each are essential in the research process and deficiency in any of: research report writing, the quantitative analyses/reports on which they are based, and the capture and manipulation of the electronic data on which both are founded, will detract from the utility of these data.

Native English speakers or proficient non-native speakers. However, much use of text formatting is designed to guide the user to what is considered important or potentially problematic.

Assumptions Readers have a minimum expertize in each of analytic, statistical, and computer techniques to understand the trade-off’s inherent in, and the complexity of, producing a version of the CPP data for ready analysis on a Personal Computer.

Major problems in accessing/using CPP data for analytic purposes as found in 2001:

Institutional Nominally in the public domain at NARA available for a ‘small’ fee (>$500).

Electronic data Eighty column punch-card record format stored (EBCDIC) on reel-reel tape media (32) for use on an IBM mainframe computer. Undocumented versions in ASCII format (61 ASCII) on CD-ROM available on an ad-hoc basis from NICHD.

Documentation 6,000 pages on 29x microfiche (up to 98 pages - 7 by 14 - on each microfiche), 75 microfiche total, compiled/written ten years after data collection ended by persons not involved in the design or conduct of the CPP. Dispersed, ad-hoc hard-copies of original documentation in the possession of individuals in Government, private research corporations, and Universities.

Size and complexity Helped by deliberate redundancy/repetition in all new documentation.

Media transfer: NARA reel-reel-tape data (32 tapes) to ASCII

nara media transfer.pdf

Data The 32 public use IBM/EBCDIC data files were successfully written in PC ASCII format to 2 CD-ROM disks by NARA (<1Mb-0.5 Gb, 0.8 Gb total, nara media transfer.pdf).

There are two large datasets/files: MDF0378.ASC 501Mb (101 Mb zipped)
VARFILE.ASC 95 Mb (15 Mb zipped)

The other 30 'work' ASCII datasets *.ASC total 99Mb (0.1-24Mb, 15Mb zipped)

Filenames Each of the ASCII datasets corresponds to one of the reel-reel tapes, but the ASCII datasets were renamed by NARA staff .pdf in DOS 8.3 format (cd insert)

File format Most 25/32 (The most notable exception is VARFILE.ASC), are in 80-column punchcard format i.e. there may be more than one record/line for an individual/case in each dataset.

File content VARFILE.ASC is the most important dataset of 7 which was manipulated to one record per case originally. It contains a selection of over 1200 variables.

MDF0378.ASC contains all 6.1 million card records from March 1978, 6700 variables.

The other 30 files, the 'work' files, are subsets of the data pertaining to certain topics.

CPP electronic data: ASCII; SAS, SPSS, and STATA system data sets.

Some of the CPP data is supplied as:	1.) ASCII data from NARA	*.ASC
or in compressed format (Winzip 8.1)	2.) SAS system datasets,	*.SAS7BDAT
	3.) SPSS portable datasets	*.POR
	4.) STATA system datasets	*.DTA
	5.) ASCII data from NICHD	*.

1. SAS\Windows 8.2e is the program used here to read the NICHD ASCII files into a system format. The basis of the programs are: fixed-column conditional input using the hold pointer @ by card, match-merging the cards by NINDB for each data collection form and/or group of forms. They have been written and formatted for clarity and ease of modification for direct input into other software packages using 'find and replace' or regular expressions, not elegance or parsimony.

2. SPSS system datasets were written from SAS using DBMSENGINES 7.0 (DBSPSSX/PR), opened in SPSS 11.0, and then saved as SPSS portable files using SPSS 11.0.

3. SPSS has a distinct advantage over SAS in storing variable value labels within the system data set. In SAS, these values are kept with separate syntax files, applying them using PROC FORMAT. SPSS limits variable names to 8 characters.

Why these formats? While transfer to ASCII files on CD-ROM makes the CPP accessible to many more researchers, they still must read it into a statistical or database software package to generate descriptive reports and inferential statistics for analytic purposes, the two most used in this research genre are SAS and SPSS. Database packages do not compute the inferential statistics necessary for advanced statistical modeling, and spreadsheets have storage and memory issues in these large (# of records 'long' - 6.1 million in MDF0378.ASC, and # of variables 'wide' - >6,700).

Creating these formats. SAS is the de facto 'gold standard', and its chief advantage is its data management capability without resort to higher order programming like C or Basic. SQL functionality is included in PROC SQL. The trade-off is the steep learning curve and relative complexity compared to SPSS' data step. The statistics generated by both are, for most purposes, equivalent. SPSS switched its efforts to utilizing Windows GUI capabilities earlier, and does not retain the mainframe, command interpreter 'feel' still noticeable in SAS. STATA is a more recent arrival and may overtake both SAS and SPSS in popularity. It is still far less widely used, but it is included here because it is the basic package used for training at JHU Medical Institutions.

DBMS Engines provides the capability of writing other system data sets directly from SAS. SAS reads SPSS transport files (usually *.POR) using PROC CONVERT syntax and the menu \data\table dialog box. However, there are considerable technical difficulties in producing STATA datasets and they are supplied but must be approached carefully: datasets are limited to 2047 variables and its methods of storing numbers as 8 byte floats is causing transcription problems within DBMSENGINES and SAS.

SPSS has had the capability of directly opening post SAS 6 data sets (*.sas7bdat) since Version 10.0.05 but this author has found that it does not always handle large (wide) datasets (>1,000 variables) well and cannot properly read SAS datasets created with SAS compression option on (COMPRESS=YES).

Since SAS and SPSS now read each others system files directly the system data formats are supplied. While our aim is use on a PC, some researchers, particularly if operating in a mainframe environment other than UNIX (SAS Windows and SAS UNIX files are identical for most practical purposes and DBMS engines uses the same engine for both) may not care to, or have difficulty with, upload(ing) other system data sets.

SPSS users: punchcard records are 'GROUP NESTED' and may be read directly into SPSS (spss read in example.pdf) from ASCII files. Modify rest of SAS syntax using Find and Replace, Textpad, or any regular expression capable editor. A sample program is given for forms ped-1 and ped-2 (SPSS read in example.pdf), but most users should be able to use SAS or SPSS datasets in Windows software, without resorting to direct input from ASCII.

STATA users: STATA does not handle large datasets well: it reads the whole dataset into RAM so there must be more RAM than data set size. Also, there were considerable technical difficulties in reading data directly from ASCII into Stata and considerable technical difficulties in conversion to Stata data sets using SAS and DBMS/ENGINES' dbstata ver=64 engine. Sometimes there are memory errors (set memory=###mb ###=RAM on your PC). Sometimes there is a 'floating point exception error' in SAS/DBMS ENGINES, and sometimes SAS/ENGINES will write a Stata data set, but it is incorrect when opened in Stata. Therefore, it may be better for Stata users to select their variables from the documentation and create their own dataset with only the variables they want. Example programs and data dictionaries (varnumv.sta stataex.txt)) and technical tips (STATA help .pdf) are supplied. The infix statement is used for fixed column input. Take particular care in checking the NINDB number, which is > than the default 8 and precision may be lost unless int or long is used - unacceptable in an ID.

Variable naming convention

The large (>6700) number of variables and SPSS' limitation of 8 character variable names makes it difficult to name variables in a manner which informs the reader of their content. However, incorporating information in the name can save considerable time in finding and using them.

Since these data were collected in 80-column punchcard record format the field number in the documented 'definition of codes' and punch card number has been used as the basis for the variable name, and the explanation of that name i.e. how to determine what information/variable it contains. In modern jargon, the 'definition of codes' is a combination data dictionary and codebook. 'Definition of codes'. The definition of codes is part of the microfiche/pdf documentation (section II,A-J .pdf), and it also exists as a much xeroxed hardcopy impact typed in 1978 after creation of the electronic Master Data File.

Field is **NOT** the same as 'variable', more aptly line/paragraph in definition of codes.

Each variable name, limited to 8 characters,

1. Character 1: F for field F#######

2. Character 2 and 3: field number from the definition of codes F23#####

3. Characters 5 through 8: punchcard number - 5 is the card number in the series for that form and they are not always consecutive (see CPPASCII.CON) 678 is the form. F23#5678

4. Character 4 is usually an underscore, acting as a visual break and placer. F23_5678

Thus, f23_5678 would read 'field 23 of the definition of codes for card 5678', which is card 5 for data collection form 678, which is usually form 78 of series 6.

However, some forms have different digits than their title number in 678 but they are unique.

Some 'forms' are created items - there were no physical data collection forms or paper punchcards but the

FUNDAMENTAL CONCEPT OF THE FIXED 80 COLUMN DATA STRUCTURE IS THAT COLUMNS 1-5 SPECIFY WHAT INFORMATION IS IN COLUMNS 66-70 AND COLUMNS 6-14 SPECIFY WHO IT WAS COLLECTED ON.

This may be looked up in the definition of codes for that form and its punchcards either on the microfiche/pdf or in the 1978 hardcopy.

In some cases, usually the first field in a punchcard F1, the field refers to several items of information in a previous punchcard for that form. For example, F1 of card 2 for form 678 could refer to fields 1 through 5 of card 1 for form 678. As we would put it, there are five variables in F1 for card 2 for 678.

This is typical for dates mmddyy which are one field in the documentation but we split into 3 (mm dd yy) variables; and age, sex, race collected on each card for a form.

In these instances the fourth character F##4####, usually an underscore,
is used and F1_2678 is broken down into f1a/b/c/d/e2678 or a date

F##a2678 F##b2678 F##c2678

Of course, any user may create their own variable name if they read data directly from ASCII format or rename variables, but the naming convention tells both where the variable's data was read from and directions to its explanation in the definition of codes without reference to cumbersome indexes in the documentation.

THIS NAMING CONVENTION HAS NOT BEEN USED FOR THE COMPENDIUM 'VARFILE' WHICH IS TREATED SEPARATELY (see varfile users guide.pdf). The varfile variables have been numbered according to the order of their definition/description in hardcopy documentation (1972) and not their order in the data set (column-order). NOTE THAT NEITHER THE VARFILE VARIABLE NAME OR THE 1972 DOCUMENTATION REFER EXPLICITLY TO THE FORM or PUNCHCARD. This definitive information is found in the Varfile 'fiche/pdf documentation. However, it is clear in most cases and (variable file contents .pdf) helps.

Compact Disc case inserts, covers, and jewel case labels:

This is the first attempt, and it will be much improved in the next version, including printing on a sharper laser printer. The aim is to use the cd jewel case inserts as a ready reference for all and a 'quick and dirty' first-time user's guide, especially for those already using CPP data. Quick and dirty means there are exceptions to every rule and convention in CPP data which is too complex to properly describe in a couple of pages - hence the 6,000 pages of dicumentation. Two loose inserts are supplied, if a third is necessary it may be better to produce a booklet.

The inserts orient the user and emphasize what must be known to use these data.

Glossary of important terms and names:

@
holds the pointer in SAS for conditional input:, here by card number

ACPP
Archives CPP - data as transferred from mainframe to PC format by National Archives and Records Admin., Center for Electronic Records

ASCII
A standard plain text format for PC electronic files with no formatting or database aspects to electronic storage.

Definition of codes, 1978
CPP hardcopy document
In modern jargon a combination data dictionary and codebook finalized in April 1978 after the final version of the CPP electronic data was created (MDF0378.ASC). It was reformatted and incorporated into the microfiche documentation. Pre PC and word processing, it exists in hardcopy only (parts have been word-processed), often old xerox copies several generations removed from the original. This hardcopy is probably the most useful and definitive document in understanding the CPP electronic data. It did not scan for OCR (see jewel case insert).

EBCDIC
E .. BINARY CODE ... DIGITAL INFORMATION ..C??????

fiche
Microfiche. Reduced scale photographs of documents common before the advent of electronic documents. CPP documentation (about 6,000 pages) was microfiched at 29x on 75 microfiche. They were transferred to Adobe portable document files (0001 through 0073,0073a, 00773b .pdf (CD 1)) and edited (CD 2 subdirectory 4) versions are in progress.(see jewel case inside back)

Field
Numbered paragraph in ‘definition of codes’ which may have more than one item of information/variable (see jewel case insert).

Forms
Standardized instruments and schedules used for data collection in the CPP. Not to be confused with normal forms - the principles of modern database design. CPP forms came in series and are digits 2-4 of each punchcard record. They were revised often and the version/revision is digit 5 of each punchcard record. A copy of each form is supposedly in the microfiche documentation. Many did not photograph well and are blurred.

Master Data File
MDF0378P.ASC
All CPP data; 6.1 million punchcard records, are in the final version of this electronic file created April 1978
At least 6700 variables.

merging or match-merging
linking or combining datasets with common individuals to add variables only. Can be a complex process due to duplicate individuals (not keyed/indexed) and individuals only in one dataset. Inexperienced analysts are advised to master merging two datasets before attempting 3 or more in one step. Not the same as concatenating or appending - adding individuals (sometimes

variables too).

NARA, CER
National Archives and Records Administration, Center for Electronic Records, College Park, Maryland.

NCPP
In this context, the version of the CPP ASCII data supplied by Dr. Mark Klebanov, NICHD, NIH on two CD-ROM disks. It is the ACPP IBM/EBCDIC master data file mdf0378.asc and/or work files either subsetted/converted into 61 ASCII datasets by topic or form or unchanged. NCPP data was used as the basis for SAS input programming because it was obtained first, removes a step, and was used by others outside JHU (see jewel case insert).

NINDB (id #)
ID number assigned when the CPP's institutional home was the National Institute for Neurological Diseases and Blindness (sometimes Stroke - NINDS). It is 7 digits for the mother/family, 8 digits for the enrolled pregnancy, 9 for the child(ren). Context supplies which is apropos but the varying length of the key/index causes problems in match-merging. See CD insert for more.

Normal forms
The rules of modern database design and best practice if followed carefully. Any analyst/data manager is warned that the original CPP data breaks them except for the compendium variable file and a few others.

pdf
Well known Adobe Acrobat portable document format legible with the freely available Acrobat Reader.

punch-card
Physical paper card with holes punched according to values of a variable used to input data into computers before the advent of direct electronic keyboard input.

regular expressions
Sophisticated wildcards for searching and manipulating text and text file contents. Most readers will be familiar with the subset used in DOS * ?.

SAS
Statistical Analysis System. De facto standard software package in health research. for manipulating large datasets. SAS Institute, Cary, North Caroloina. Version 8.2e, Windows..

SPSS
Most used software package for analysis in psychosocial and related health research. Less data management features than SAS. Conceptual Software Inc., Houston, Texas?. Version 11.0, Windows.

STATA
Software package used at JHMI for training. Version 7.

VARFILE.ASC
A useful dataset containing a compendium/selection of 12000 variables from the total 6,700. Sole original dataset with substantial information about mother and child from the womb through age 8. (see jewel case insert)
**** It has one record per child or pregnancy. ****

Work files
Subsets of CPP data (30) from the electronic master file used on an ongoing basis for analysis. The original CPP documentation on 'fiche

recommends using these files, not the Master File, for analyses, but they are not compared here. NARA CER staff used the DOS 8.3 naming convention for the ASCII versions so these filenames are shorter than the IBM mainframe tape/filenames (see jewel case insert).

There is a bibliography of publications from the CPP maintained by Dr. Matthew Longnecker, EIS, NIH at

http://dir.niehs.nih.gov/direb/cpp/pubs_cpp4.htm

Microfiche 73a and 73b contain a CPP bibliography to the mid 1980s.