Documentation for Epi-C

EpiC - The EpiData command application

Version 3.1.2

Programmed by M.Bruus. Design: JM.Lauritsen & M.Bruus

Revision history: 3.1.2 (fixed problem with export to Stata 8, nov 2006)

Functions included so far:

? Export datafile
? Import
? Consistency check
? Revise datafile from revised QES-file
? Count records by field
? Pack
? zip/unzip
? encrypt/decrypt
? validate dublicate files
? append/merge

EpiC is a utility for some of the functions and tools in EpiData. EpiC can be run from a command prompt or from a command (or batch) file, making it possible to automate repetitive tasks.

EpiC interacts very little with the user and is considered a tool for the advanced user.

After perfoming a function, EpiC returns an errorcode. The errorcode 0 (null) means that the function was performed without errors. The meaning of all other errorcodes is shown in the bottom of this document or by typing **EpiC codes**. The errorcodes can be used in commandfiles (bat-files) to give feed-back to the user as well as in check files.

```
FELT1
  AFTER ENTRY
    DEFINE tmpN ####
    IF felt1=1 THEN
      EXECUTE EpiC “zip myzip.zip path *.*” WAIT noHIDE
      tmpN=RESULTVALUE
      IF tmpN<>0 THEN
        HELP "Problems with EpiC\nErrorcode=@tmpN"
      ENDIF
    ENDIF
  END
END
```

All functions in EpiC writes status and errorcodes to a log file. See below how to find the log file or specify a name of a log file.

Conventions in this document: all commands and options inclosed in square brackets [ ] are optional. **e[xport]** means that you can write **e** or **export** to specify an export-command. Commands and options written in *italic* are not to be entered litteraly. *Inputfilename* means that you enter a name for an input file, not that you type "inputfilename". Vertical line "|" separates two or more alternative options. UPPER|LOWER means that you can type the option UPPER *or* the options LOWER.

EXPORT

Syntax from Command-prompt :

```
  EpiC  e[xport] exporttype inputfilename outputfilename|*
        [FROM=first_rec_no] [TO=last_rec_no] [DELETED] [REPLACE]
        [UPPER|LOWER] [DELIM=delimiter] [NONAMES] [FILTER="filter"] [LOG="logfilename"]
```

First parameter after EpiC is the command or function to be executed.
Specify e or export to do an export. Exporttype, inputfilename and outputfilename must be given and in that order. The options are not required and they can be in any order.

Parameters for export:

exporttype Can be
TXT or TEXT to export to textfiles
DBF, DBASE or DBASE3 or DBASEIII to export to dBase III format
EXCEL or XLS to export to Excel 2.1 format
DTA, STATA, STATA4, STATA5, STATA6 or STATA7 to export to Stata
STATA or DTA without a number will export to Stata ver. 6
SPSS or SPS to export to SPSS
SAS to export to SAS
EPIDATA or EPD or REC to export to a new EpiData datafile

Inputfilename Must be a rec-file. The extension .rec can be omitted.

Outputfile *Outputfilename* specifies the name of the exported file.
If * is used as name then the outputfile will have the same name as the inputfile, but with a relevant, new extension (.new.rec for export to EpiData)

FROM FROM=10 specifies that records #1-#9 are not exported.
Default value is 1.

TO TO=90 specifies that only records up to and including record #90 are exported. Default value equals the number of records in the inputfile.

DELETED specifies that records marked as deleted in the inputfile will be exported.

REPLACE specifies that an existing file with the same name as the outputfile can be overwritten. If outputfile allready exists and REPLACE is not specified, then export aborts with the errorcode 63.

UPPER UPPER and LOWER specifies the case of the fieldnames in a resulting Stata file. Default is that fieldnames are exported unchanged from the inputfile.

DELIM specifies the delimiter between fields in a resulting textfile. Default is comma (,)

NONAMES specifies that fieldnames will not be exported as the first line in an export to a text-file.

FILTER Use filter to export only records that meet a specified logical condition.

LOG LOG="*logfilename*" specifies the file where output from EpiC is saved. Default is a file called "EpiC.log" which is saved in the same directory as the inputfile. New output is appended to the file.

Examples:

EpiC e stata allaes *
Exports allaes.rec in the same directory as epic.exe to a new Stata 6 file with the filename allaes.dta. All records in allaes.rec are exported.

EpiC e xls allaes * filter="ID=276"
Exports allaes.rec to allaes.xls. Only records where the field ID equals the value 276 are exported.

EpiC e dbf allaes * filter="AESRATER='TP'"
Exports allaes.rec to allaes.dbf. Only records where AESRATER equals 'TP' are exported.

EpiC e epd allaes "c:\new\newallaes.rec" fromrec=100 torec=199 log="c:\new\mylog.txt" WITHDELS
Exports allaes.rec to a new epidata datafile called c:\new\newallaes.rec. Only records #100-#199 are exported. Deleted records are exported too. Output from epiC is saved in the file c:\new\mylog.txt.

EpiC e epd allaes * delim=TAB replace
Exports allaes.rec to allaes.txt. Delimiter will be a tab character. Note TAB in capital. If existing allaes.txt will be replaced.

IMPORT

EpiC can import textfiles (delimited/fixed), dBase III files and Stata files (versions 4-8).

***General** syntax for import:*

EpiC i[mport] *importtype inputfilename outputfilename*|* *options*

Importtype can be TXT or TEXT for import of textfiles
DBF or DBASE for import of dBase III files
STATA for import of Stata files

Inputfilename can be any filename. If no extension is shown then .txt, .dbf or .dta is added to the name depending on the importtype.

outputfilename is the name of the resulting EpiData datafile. Extension .rec is added. If output file is specified as * then the resulting datafile will have the same name as the inputfile, but with the extension .rec

Importtype, inputfilename and outputfilename are required parameters and must be given in that order. Options are not required and they can have any order.

*Syntax for import of **textfiles**:*

```
EpiC i[mport] TXT|TEXT inputfilename  outputfilename|*   [QES=qesfilename]
    [delim=delimiter [q[uotes]=all|text|none]]  [REPLACE]
    [ignorefirst] [firstword] [date="dateformat"] [LOG="logfilename"]
```

qesfilename is the name of the EpiData QES-file that acts as mask for the import. If not specified then qes-file is assumed to have the same name as the inputfile, but with the extension .qes.

delim specifies the field delimiter in the text-file. Delimiter can be any character. Write delim="TAB" or delim="SPACE" to use tabulatorcharacter or spacecharacter as delimiter.
If no delimiter is specified then the textfile is assumed to be in fixed length format.

quotes specifies is all fields, only textfields or no fields in the textfile is surrounded by double quotes. Ignored in fixed length textfiles. Default if not specified is quotes=none

replace specifies that an existing file with the same name as the outputfile can be overwritten. If outputfile allready exists and REPLACE is not specified, then export aborts with the errorcode 63.

Ignorefirst If specified then the first line in the textfile is ignored. Can be used to skip a line with fieldnames. If not specified then first line in the textfile is imported.

firstword specifies that the new datafile must be created from the qes-file using first word as fieldname. If not present then automatic fieldnaming is used.

date specifies the format of dates in the textfile. Examples of dateformat:
date=dd/mm/yyyy (dates 1-9 are prefixed with 0, 4 digit year)
date=mddyy (months 1-9 are not prefixed with 0, 2 digit year, no separatorcharacter)
date=yymmdd (2 digit years, no separatorcharacter)
Default is date=ddmmyyyy.

Examples of textfile import:

```
EpiC i TXT mytxtdata.txt resultfile.rec qes=themask.qes delim=; q=text ignorefirst date=yymmdd
```

Imports data from the textfile "mytxtdata.txt", which has semicolon between the variables (delim=;), doublequotes around text variables (q=text), variablenames in the first line (ignorefirst) and date variables has the format yymmdd. Import is based on the qes-file "themask.qes", which supplies the EpiData field names and field types. Fieldnaming in the qes-file is automatic ("firstword" is not specified). The resulting EpiData datafile is saved as "resultfile.rec".

EpiC import TXT samples.txt * log="samples.log"

Import data from the fixed length (delim not specified) textfile "samples.txt", which does not have fieldnames in the first line (ignorefirst is not specified). Dates are formatted as ddmmyyyy (which is the default format in EpiC). A qes-file named "samples.qes" is used as mask and it specifies field names and field types using automatic fieldnaming (firstword not specified). The resulting datafile is saved as "samples.rec". Progress of the import is logged in the file "samples.log".

*Syntax of import of **dBase III** files:*

EpiC i[mport] dBase *inputfilename outputfilename*|* [date=dmy|mdy] [REPLACE] [LOG="*logfilename*"]

inputfilename Name of dBase file to import. Extension .dbf will be added if not present.

outputfilename Name of resulting EpiData datafile. If outputfilename is * then the outputfilename will be set to same as inputfile, but with extension .rec.

date Specifies if resulting EpiData datafile presents dates in the dBase file as dd/mm/yyyy or mm/dd/yyyy. Default if date is not specified is date=dmy.

replace specifies that an existing file with the same name as the outputfile can be overwritten. If outputfile allready exists and REPLACE is not specified, then export aborts with the errorcode 63.

Examples of import from dBase III files:

EpiC i dBase heights *

Imports the dBase III file "heights.dbf" and saves the EpiData datafile as "heights.rec". Dates in the resulting EpiData datafile will be formatted as european dates (dd/mm/yyyy).

EpiC import dbase weights.dbf newweights.rec date=mdy log="weights.log"

Imports the dBase III file "weights.dbf" and saves the EpiData datafile as "newweights.rec", which will have dates formatted as american dates (mm/dd/yyyy). Logging of the import is done in the file "weights.log".

*Syntax of import of **Stata** files:*

EpiC i[mport] stata *inputfilename outputfilename*|* [REPLACE] [LOG="*logfilename*"]

inputfilename Name of Stata file to import. Extension .dta will be added if not present. Stata version 4, 5, 6 and 7 can be imported.

outputfilename Name of resulting EpiData datafile. If set to * then the outputfilename will be set to same as inputfile, but with extension .rec.

replace specifies that an existing file with the same name as the outputfile can be overwritten. If outputfile allready exists and REPLACE is not specified, then export aborts with the errorcode 63.

CONSISTENCY CHECK

Syntax from Command prompt:

EpiC cc *inputfilename* [*name of file with consistency checks*] [LOG="*logfilename*"]

First parameter after EpiC is the command or function to be executed.
Specify cc or CONSISTENCYCHECK to do a consistency check. The command must be the first parameter.

Second parameter is the filename of the EpiData datafile to be checked. Is fileextension ".rec" is not part of the filename, it will be added.

Name of file with the consistency checkblock can be added as the third parameter. If a file is not specified – or the file is specified as * – then EpiC will use the CHK-file with the same name as the inputfile.

Name of logfile can be added as the fourth (or third) parameter. The log-file will contain the result of the consistency checks or errors that occured during the checks. If no log-filename is specified then a log-file with the name "EpiC.log" and the same directory as the inputfile will be used.

See EpiData helpfile for complete details on the syntax of the consistency checkblock.

Examples are:

```
CONSISTENCYBLOCK
  REPORT ID
  CHECK "V1 is unusually large " (V1>112)
  CHECK "ID-numbers are not identical" ID1=ID2
  CHECK "Ranges" CHECKRANGE
  CHECK "Legal"  CHECKLEGALv
  CHECK "MustEnter" CHECKMUSTENTER
END
```

If you wish to test for combined range and legal use:

```
CHECK "Range or legal" CHECKRANGELEGAL
```

e.g. if you defined range as 0-50, plus legal 99. Otherwise all the 99 would be rejected by the CHECKRANGE and the allowed values in range 0-50 would be rejected by CHECKLEGAL

REVISE DATAFILE FROM REVISED QES-FILE

Syntax from Command-prompt :

EpiC rev[ise] *qes_filename datafilename*|* AUTO|FIRST [FORCE] [LOG="*logfilename*"]

REV[ise] command, qes_filename and datafilename must be given and in that order. If the data file has the same name as the qes-file the datafilename can be specified as *.

FORCE If FORCE is used as option, the revision will take place even if it

results in loss of data. Loss of data can occur if the qes-file sets
a lower number of decimals in a number than is found in the datafile.
Loss of data also occurs if all data of a field are not used because
the field's name is not found in the qes-file. In the last case it
will show in the log-file what fields are skipped.
If FORCE is not present, the revision will be stopped if there is a
risk of loss of data

AUTO creates the revised datafile from the qes-file using automatic fieldnaming. Either AUTO or FIRST must be specified.

FIRST creates the revised datafile from the qes-file using "first word" fieldnaming. Either AUTO or FIRST must be specified.

LOG LOG="*logfilename*" specifies the file where output from EpiC is saved. Default is a file called "EpiC.log" which is saved in the same directory as the inputfile. New output is appended to the file.

COUNT RECORDS BY FIELD

Syntax from Command-prompt :

EpiC count *fieldname file1* [*file2* [*file3* [*file4*....]]] [LOG="*logfilename*"]
or
EpiC count *fieldname* FILELIST=*filelist_filename* [LOG="*logfilename"*]

Name of field to count by must be the second parameter. At least one file to examine must be present. A maximum of 25 files can be examined at the same time. If fileextension ".rec" is not part of the filename(s), it will be added.

In the first syntax the files to compare are listed as parameters. In the second syntax the names of the files to compare are saved in a textfile with one – and only one – filename per line.

The resulting count table is saved in the log file.

PACK

Syntax from Command-prompt :

EpiC pack *filename*[.rec] [LOG="*logfilename*"]

Pack permanently deletes all records in the specified datafile.

ZIP

The zip command makes pkzip compatible packed files. You can unzip them with EpiC, pkzip or any other programme which handles standard format zip files.

Syntax from Command-prompt :

EpiC zip *zipfilename* [path] [update|replace] [date] *files [files [files...]]*

Zips the files specified by *files* into a zip-file.

ZIPFILENAME Name of the zip-file to be created or updated

PATH Recurse subdirectories and add relative directory information to zip-file

UPDATE If zipfile exists then *files* are added to the zipfile (default)

DATE Adds date-stamp to filename

REPLACE If zipfile exists then it will be deleted and replaced by a new zip-file containing *files*

FILES Specification of files to zip. Examples: c:\myfiles\*.* d:\music\*.waw c:\myfiles\rtf\qe?.*
Several filespecifications can be added

UNZIP

Syntax from Command-prompt :

EpiC unzip *zipfilename destdir* [replace]

Unzips the files in the zip-file *zipfilename* to the directory *destdir*.

ZIPFILENAME Name of the zip-file to unzip (include path)

DESTDIR Destination directory (e.g. c:\WINDOWS\TEMP\NEWDIR)
If destination directory does not exist, it will be created.
If zip-file contains subdirectories and path-information, these will be created.

REPLACE Existing files with the same names as the files being unzipped will be overwritten.

ENCRYPT

Encryption can be made to any file.
Encryption uses http://csrc.nist.gov/CryptoToolkit/aes/rijndael Rijndael/AES strong encryption algoritm.
Files made with earlier versions of EpiC used the blowfish Http://www.counterpane.com/blowfish.html encryption algoritm and should be decrypted with version 1.2 of EpiC available from http://www.epidata.dk

Once encrypted the keyword cannot be guessed. The keyword is NOT a pkzip password.

Please note that once encrypted there is no way to guess the key !! Neither by us or any sort of “reverse guess” procedure.

Syntax from Command-prompt :

EpiC en[crypt] *filename key* | ? [replace]

Makes an encrypted copy of the file *filename*. The encrypted file will have the name *filename.key*.

FILENAME Any file including path

KEY	A word or a phrase of minimum 6 characters. Enclose in double quotes if it contains spaces (e.g. "My password"). If key is ? or "key=?" then EpiC will ask the user to enter and reenter the keyphrase.
REPLACE	If *filename*.key already exists it will be overwritten

DECRYPT

Syntax from command-prompt :

EpiC de[crypt] *filename key* **|** ? [replace]

Makes an decrypted copy of the cryptated file *filename*. The cryptated file must have the extension .key. The decrypted file will have the same name as *filename* but without .key (e.g.: myfile.zip.key will be decrypted to myfile.zip).

FILENAME	Any cryptated file including path and with the extension .key
KEY	A word or a frase of minimum 6 characters. Enclose in double quotes if it contains spaces (e.g. "My password"). If key is ? or "key=?" then EpiC will ask the user to enter the keyphrase.
REPLACE	If resulting file already exists it will be overwritten

VALIDATE DUPLICATE FILES

Syntax from command-prompt:

EpiC v[alidate] file1[.rec] file2[.rec] [key1 [key2 [key3]]] [DEL] [NOTEXT] [CASE] [TYPEDIF] [NOMISS2] [LOG="logfilename"]

Compares file1 with file2 and makes a report on the differences in the log-file. File1 and file2 must be EpiData rec-files. Extension .rec can be omitted.

KEY1, KEY2, KEY3	Optional fieldnames that are used to match a record in file1 to a record in file2. If more than one key is given then records are matched on all fields. If omitted then records are matched by sequential record number.
DEL	If present, then records marked as deleted are included in the process. Default is to ignore deleted record.
NOTEXT	If present then text-fields (alpha, uppercase alpha, encrypted fields) are ignored.
CASE	If present then differences in letter-case in text-fields are ignored
TYPEDIF	Ignore differences in field type. Default is to report differences in field type.
NOMISS2	Ignore missing records in file2. Use this option if file1 is a re-entered subset or sample of the data in file 2.

APPEND

Syntax from command-prompt:

```
EpiC app[end] fileA fileB fileC [KEYA=key] [KEYB=key] [USEALLFROMB]
```

FileA and fileB are names of existing rec-files. The two files are appended and saved as a new rec-file with the name specified in fileC.

KEYA Keyword for fileA if the rec-file contains encrypted fields

KEYB Keyword for fileB if the rec-file contains encrypted fields

USEALLFROMB Add all fields from fileB to resulting file. Default is that only fields common to fileA and fileB are saved in the resulting file.

If fileA or fileB is encrypted, then resulting rec-file will have same key as fileA.

MERGE

Syntax from command-prompt:

```
EpiC MER[GE] fileA fileB fileC KEY=keyfield [MERGEALLB] [COMMON=a|b] [KEYA=xxx] [KEYB=xxx]
```

FileA and fileB are names of existing rec-files. The two files are merged into a new rec-file with the name specified in fileC.

KEY Name of keyfield that is common to fileA and fileB

MERGEALLB All records from fileB are added to fileC. Default is that only records with a common keyfield are added to fileC

COMMON If different values are found in fileA and fileB in fields with common fieldnames, then common=A will use the value in fileA (default). Common=B will use the value in fileB.

KEYA Keyword for fileA if the rec-file contains encrypted fields

KEYB Keyword for fileB if the rec-file contains encrypted fields

Acknowledgements:

Blowfish: Http://www.counterpane.com/blowfish.html
Component for Delphi for blowfish and Rijndael by: D.Barton http://www.scramdisk.clara.net/

Zip file functions are based on the VCLZIP component
kindly donated to EpiData, see http://www.bigfoot.com/~vclzip

---

ERRORCODES

0	No error
1	Unknown file / file not found
2	Assertfile not found (in consistency check)
3	Out of memory
4	Error opening datafile
5	Error in checkfile
6	Datafile has no records
7	Error opening file containing consistency checks
8	Error opening qes-file
9	Error in qes-file
10	Invalid filter (export)
11	Function failed
12	Error creating outputfile
13	Invalid value found in datafile
14	Error in consistency checks
50	Error in parameters for EpiC
51	No inputfile specified
52	No exporttype specified
53	No outputfile specified
60	Error creating new datafile (revise datafile)
61	Risk of loosing data
62	Incompatible fields found (revise datafile)
63	Outputfile allready exists and REPLACE not specified
65	AUTO or FIRST parameter missing
70	Fieldname to count by not specified
71	No inputfiles specified
80	Inputfile is not valid dBase III file
81	Inputfile is not valid Stata file
82	Too many variables in inputfile
83	Error in date parameter (import textfiles)
84	String variables are too long (max 80 chars. allowed)
90	Error in zip or unzip operation
91	Error in zip/unzip parameters
92	No files were zipped
100	Encryption key too short
101	Inputfile not found
102	Outputfile already exists
103	Error opening inputfile
104	Can't create outputfile
105	Encryption/decryption error
106	Incorrect decryption key