

SOEPMENU:
A Menu-Driven Stata/SE Interface for Accessing
the German Socio-Economic Panel

John P. Haisken-DeNew¹ (<http://www.SOEPMENU.de>)

This Documentation Version: Feb 2005
First Documentation Version: Dec 2004

Abstract: This paper outlines a panel data retrieval program written for Stata/SE, which allows easier accessing of the German Socio-Economic Panel Data set. Using a drop-down menu system, the researcher selects variables from any and all available years of the panel. The data is automatically retrieved and merged to form a rectangular “wide file”. The wide file is transposed to form a “long file”, which can be directly used by the Stata panel estimators. The system implements modular data cleaning programs called plugins.

Keywords: Panel data storage and retrieval, SOEP, Germany
JEL: C81, C87, C23

¹ The author wishes to thank the following persons: W. Paul DeNew, Michael R. Veall, Klaus F. Zimmermann, Gert G. Wagner, Christoph M. Schmidt, Peter Szmyt, Joachim R. Frick, Markus Panneneberg, Ingo Sieber, Ingo Geishecker, Mathias Sinning, Patrick Dross, Christopher Baum, Stephen Jenkins, Goetz Rohwer, William Gould, Kevin Crow.

1. Introduction

Accessing the German Socio-Economic Panel Data (SOEP) set has often in the past been a challenge for many social scientists and students alike. The SOEP is a collection of data files currently numbering over 240 in the regular distribution from the DIW Berlin. Data contain information collected from the longest running European household panel survey, including information at the person, household, and spell level. In addition there have been numerous additional data files provided which have been generated by the SOEP Group at the DIW Berlin containing derived variables or user-friendly versions of biography data. See Haisken-DeNew and Frick (2002) for details.

In order to deal with the complexity of this very large panel data set, an internet based tool called SOEPinfo was established by John Haisken-DeNew and Ingo Sieber in the SOEP Group, already in 1996 in a rudimentary form². This tool allowed users to “collect” SOEP variables by clicking on them with a mouse, and once variables were “added to basket”, actions could be run on the variables, such as displaying pre-generated frequencies, or jumping to the exact place in the questionnaire where the variables could be found. Alternatively, one could automatically create a command file in Stata, SAS or SPSS to retrieve the variables “collected in the basket”. Retrieving the data outright contravenes data security laws in Germany and is therefore not viable. However creating the command files to retrieve data locally installed on the user’s own machine is indeed legal and efficient. This allowed the user to run a data retrieval on his own local machine and create a rectangularized data file in “wide format”, such that there would be a person or household as the unit of analysis and additional years of information would be stored as additional columns in a spreadsheet.

An integral part of SOEPinfo has always been the item-correspondence list, i.e. a list of items, such as “Wage” or “Unemployed Status” which contain lists of actual variable names corresponding the variables in a particular year. For better or for worse, the naming scheme of variables in the SOEP reflects essentially the order of the particular question a particular questionnaire in a particular year. For instance the actual variable name AP3301 refers to personal gross wage in 1984. It simply happens to be the first part (01) of question 33 in the person (P) questionnaire in 1984 (or year A). Clearly, questionnaires evolve over time and the particular position of a question does not remain constant. Thus, there is no guarantee whatsoever that the same “wage” question in the following year follows any kind of patterns similar to the previous year. Indeed the variable in 1985 (year B) is named BP4301. This variable naming scheme has the advantage of immediately identifying to the user where exactly the question can be found in the survey and whether it was a person or a household head answering on behalf of the entire household, who answered the question. On the other hand, when using many waves of information in a panel setting, keeping variables straight becomes very tedious.

The SOEP is of considerable interest to social scientists as it allows researchers to examine the same individuals over as many as 20 years using the most current panel estimation techniques available in statistical packages such as Stata, SAS and SPSS. However, to use the advanced techniques, all packages require the data to be stored in a different manner than “wide format” called “long format”. This implies transposing the wide data set to a unit of analysis based on person-years or household-years. There are no longer variable names with year indicators attached to them such as WAGE2000 or WAGE2001, but just simply WAGE with an

² Since 2003 SOEPinfo has been maintained entirely by Ingo Sieber at the DIW (SOEP).

additional indicator `YEAR` indicating the specific year that the information refers to. Thus each row in the data set refers to a person-year or household-year observation, “making the data sets markedly narrower (fewer variables) and at the same time much longer (more observations), illustrated in Table 1.

Table 1: Wide versus Long Format

Wide Format				Long Format		
PERSNR	WAGE2000	WAGE2001	WAGE2002	PERSNR	YEAR	WAGE
100	4000	4050	4100	100	2000	4000
101	-2	-2	400	100	2001	4050
102	1000	1000	1200	100	2002	4100
				101	2000	-2
				101	2001	-2
				101	2002	400
				102	2000	1000
				102	2001	1000
				102	2002	1200

However, before one simply changes a data set from wide to long (in Stata, done by the “reshape” command), one must first be assured that the variables have been made time-consistent. For example, in the simple Gender question, 1=Male and 2=Female. However this coding could theoretically change over time to 0=Male and 1=Female. Thus, all variables corresponding to the particular item “Gender” must first be harmonized before any reshaping is made. In the case of wages, the analogue would be first to deflate values by a price index to create “real wages”.

Given these challenges, SOEPMENU came into being as a method of allowing users to retrieve data directly from their own locally installed data but providing an interactive interface, such that data could automatically be transposed from wide to long and be made time consistent in a convenient manner. More specifically, SOEPMENU is a collection of Stata add-on files, which allow users of the German Socio-Economic Panel (SOEP) to access conveniently the data. The package consists of “base” and “plugin” files.

2. Basic Features

With the SOEPMENU system, the user can open SOEP data files by drop down menus. The idea behind the tool is that because of the intrinsically longitudinal nature of the data, one is interested NOT in retrieving a variable in a particular wave, but rather in retrieving the variable for several waves, i.e. an item-correspondence list. An additional complicating factor is that in the SOEP, the variable naming algorithm used reflects the order of the questions in the questionnaire, NOT reflecting any particular content. This obviously implies a changing variable name over time. Other data sets such as the BHPS (British Household Panel) cleverly use content-indicative names. Thus, if one opens a SOEP data file and one finds a variable of interest, one clicks on the variable and information for the entire item correspondence is also collected and added to a "soep project". Straightforwardly, the object is to collect items and save them into the project, allowing an automatic data retrieval.

SOEPMENU functions because Stata provides certain key commands or features in its command language and user interface. SOEPMENU requires:

- Stata SE 8.2. The internal limits of Stata Intercooled are insufficient.
- the existence of variable characteristics: Stata command `char`
- the existence of variable notes: Stata command `notes`
- the existence of drop-down menus: Stata command `window menu`
- the existence of dialog boxes: Stata command `window dialog`
- being able to buffer info in the global-space: Stata command `global`
- being able to deal with long strings in global-space: Stata command `global`

SOEPMENU allows one to create a "soep project" by various means. One can directly open a SOEP data file and add items. One can simply append other soep projects. If a SOEP data file is in memory, one can either use the drop-down-menu tool or one can create a "browse page". The drop-down-menu tool allows one to view all items found in a dataset, to search for items based on the "lookfor" command. Once items have been found, they can be added to the project. The project can be named, renamed, saved, resaved, deleted etc. The saved soep projects are saved as Stata dta files with the filename ending ".soep". If one saves projects in a modular fashion, one can create quickly full projects, e.g. `wages.soep`, `firm.soep`, and `humancapital.soep` get appended together to create `labor.soep`.

Assuming that a project is complete (the user has found all the items of interest), then by clicking on a button, the retrieval can be executed. This obviously requires that the SOEP data be located on a local or network disk that the user's computer has access to.

As a user of panel data, one would eventually like the SOEP data to be in "long" form. SOEPMENU retrieves in both manners, wide and long using the command `reshape`. This requires some standardization of variable names over time. Indeed, items in the SOEP have been given a serial number. Thus wide variables from the item number 2277 would be `ap2277x`, `bp2277x`, ..., `tp2277x`. The first letter indicates the wave [a-t], the second letter [p,h] indicates whether at the person or household level, and "x" at the end indicates a SOEPMENU variable. Thus after the `reshape` command has been executed, the first letter indicating the wave is dropped to create the long variable name for example, `p2277x`. The user can access the retrieved data at the "wide", "long" and "custom" level.

Using the SOEP, one immediately is confronted with the fact that values of variables in any particular item may change over time. For instance if 1=yes and 2=no in 1984, there is no particular reason to assume this will remain for all years following. The following year may have 1=yes 2=maybe 3=no, problematic for long data sets. Hence, using "plugins", allows the user to "clean up" these value inconsistencies. A plugin is simply an add-on in its own right with the same name as the long variable name, e.g. `p2277x.ad0`, and is executed whenever that item is selected in the retrieval. If desired, these plugins can be added, removed, reloaded, etc. For instance, one can use a plugin to deflate income measures with a price index automatically. Theoretically, there should be a plugin for each of the approximately 4,500 items, which would guarantee that all values would be consistent over time. In fact, if there is no plugin available for an item, value labels are stripped from the long variable to ensure that the user examines the item consistency over time.

The Stata files used with SOEPMENU have been preloaded with both German and English variable labels and values labels. The user can at any time switch between languages. Indeed,

one can have an English version of the SOEP on disk and create a fully German labelled retrieval³.

If for some reason, the data just created should be exported into another format, one can use several commercial tools such as StatTransfer[®] or DBMSCOPY[®], or several tools made available here. SOEPMENU allows exporting data to SPSS[®], SAS[®], LIMDEP[®], MS Excel[®] and also SQL systems. If the data is in memory, then it can be exported. In addition, the entire SOEP data set can be dumped into SQL files for import in MySQL[®]. Where variable labels and value labels have a relevant meaning in the exported format, they are kept. For instance, Limdep does not have a concept of value labels and MS Excel does not have a concept of neither value nor variable labels.

As mentioned, SOEPMENU requires a standardized naming scheme to reshape from wide to long format. This creates variable names of the following sort: `p2777x`. This is perhaps not so intuitive however SOEPMENU allows users to have the option of customizing SOEPMENU to rename SOEPMENU variables with other arbitrary names. Thus, whenever the SOEPMENU variable `p2777x` is retrieved in a “long” file, SOEPMENU automatically creates an additional “custom” file in which SOEPMENU variables have been renamed to something perhaps more intuitive such as “wage” or “labor_income”. The renaming feature is only activated if the user explicitly requests it. The user can provide custom names for variables extracted. Should a variable be found in the list of variables to be renamed, the variable is renamed according to the instructions of the user. If a SOEPMENU variable is not found in the custom list, then the variable name simply remains as it is. The exact renaming executed in the retrieval is documented in the generated retrieval DO file. Thus, there is unambiguous information as to the origins of each and every variable.

3. Installation and Maintenance

The SOEPMENU system comprises of two basic parts: (1) the SOEPMENU distribution and (2) the SOEP data pre-loaded with the item-correspondence.

The first step is to unpack the SOEPMENU archive and install the files (in exactly the directory structure in the archive) on to a disk to which you have access. The disk may be a local or a network disk. You simply need to tell Stata where to find the SOEPMENU files. Correspondingly, the whereabouts of the SOEP data files themselves must be known to SOEPMENU. The location of the SOEP data files in Stata format (`*.dta`) and the SOEPMENU add-on files (`*.ado`) must be entered into the Stata Profile, which is run every time Stata is started. Technical Appendix 1 outlines the entries to the file `profile.do` which are necessary to install the SOEPMENU commands to the Stata menuing system⁴. The Stata window title will change from “Stata/SE 8.2” to “Stata/SE 8.2 – SOEP Menu 1.1”, of course depending on the version number of Stata and SOEPMENU.

As updates of SOEPMENU become available online (in Stata: User | SOEP Menu | Help | SOEPMENU | Official Updates), the user can simply install the newer files into the same

³ Earlier versions of SOEPMENU contained an add-on which allowed one to switch between languages by redefining the labels for a particular language. This required storing the label information as variable “characteristics” and retrieving this information on the fly to redefine labels. This is no longer necessary as this language capability has since been completely integrated into Stata directly following the scheme outlined in SOEPMENU. As it has been implemented in the Stata source code, the speed gain is enormous, allowing one to switch languages almost instantaneously.

⁴ See Technical Appendix 1, where entries to the file `profile.do` are explained.

directory as the older version, thereby overwriting the older files. Additional “Plugin” files will become available as users contribute to the ever growing knowledge base. These too can be simply added to the SOEPMENU plugin directory.

References

Haisken-DeNew, John P. and Joachim R. Frick (2002) “The Desktop Companion to the German Socio-Economic Panel Study”, DIW Berlin, Germany

Table of Technical Appendices and Figures

Technical Appendix 1: Configuring SOEPMENU

Technical Appendix 2: Pre-Defined F-Keys

Technical Appendix 3: Legal Considerations

Technical Appendix 4: Overview of Files Created by SOEPMENU

Technical Appendix 5: Drop-Down Menus in SOEPMENU

Technical Appendix 6: Retrieval Options

Figure 1: Screen Dump of Menu System

Figure 2: Screen Dump of Example SOEP Project File

Figure 3: Screen Dump of SOEPMENU Project Preferences

Figure 4: Screen Dump of Possible SOEP Data Files to Open

Figure 5: Screen Dump of SMCL file created upon opening a SOEP Data File

Technical Appendix 1: Configuring SOEPMENU

Required Entries in `profile.do`

```
set memory 350m
global DB_soepath "g:/data/soep03"
global DB_SOEPMENUpath "g:/SOEPMENU"
adopath + "$DB_SOEPMENUpath/base"
adopath + "$DB_SOEPMENUpath/plugins"
set scrollbufsize 20000
set varlabelpos 14
soep
```

Technical Appendix 2: Pre-Defined F-Keys

```
F1   Open an existing SOEPMENU project file
F2   Open a SOEP wave specific data file
F3   Open a SOEP meta data file
F4   Open the Prefs Dialogue for a SOEPMENU project
F5   View/edit SOEPMENU project items
F6   Open an existing RAW file (once SOEP project has been opened)
F7   Open an existing WIDE file (once SOEP project has been opened)
F8   Open an existing LONG file (once SOEP project has been opened)
F9   View a global search results file (relevant after global search)
```

Technical Appendix 3: Legal Considerations

This product is provided only "as is". The user alone must decide whether use of this product is appropriate. No warranties or guarantees implied or intended. The author acts alone and on his own behalf. The author is not responsible for loss of data, and any resulting damages due to use of this product. Current or past employers of the author are not responsible for this product, loss of data, and any resulting damages.

SOEPMENU has been written solely by John Haisken-DeNew except for the following:

- a copied and edited version of "tosql" from Christopher Baum
Why? Because variable and value labels are not kept in the original version.
- a copied and edited version of "lookfor" from Stata corp
Why? Because items couldn't be searched in the original version.

These two external add-on files that I have used/edited from others, have their original copyright notices and can be made available to the authors of the originals at any time.

Stata, SPSS, SAS, Excel, DBMSCOPY, StatTransfer, UNIX, Limdep, EViews, R-Project, MySQL are trademarks of their respective owners and are mentioned here for informational purposes only. There is no business relationship between the author of SOEPMENU and these trademark holders.

SOEP Data are owned by the DIW Berlin, SOEP Group, who alone control access to the micro data. Please beware that data misuse offences are punishable with severe criminal and civil penalties. SOEP users are required by their SOEP contract and German federal law to protect SOEP data.

Technical Appendix 4: Overview of Files Created by SOEPMENU

For the SOEP project named "mysoepstuff", the following files are created:

Project File	mysoepstuff.soep
Item Edit File	mysoepstuff.smcl
Raw Data	mysoepstuff-raw.dta
Raw Value Labels	mysoepstuff-raw-valuelabels.do
Wide Data	mysoepstuff-wide.dta
Wide Value Labels	mysoepstuff-wide-valuelabels.do
Long Data	mysoepstuff-long.dta
Custom Data	mysoepstuff-custom.dta
Project Retrieval	mysoepstuff-data.do
Collected Plugins	mysoepstuff-plugins.do
Acknowledgements	mysoepstuff-acknowledge.smcl

File Type	Description
Project File	creates all other files.
Item Edit File	contains a list of items, for each item, a list of all variables in that item, then at the end, a list of variables, sorted by SOEP file. All project preferences used in the retrieval are listed. All plugin names are listed. This serves as useful print-ready documentation for the contents of a soep project.
Raw Data	contains the SOEP data in wide format immediately after merging all source files together, but BEFORE any data changes due to plugins.
Raw Value Labels	contains a list of all value labels BEFORE any plugins have been executed.
Wide Data	contains the raw data in wide format, AFTER the plugins have been executed. Potentially, there are additional new variables, recoded old variables etc.
Wide Value Labels	contains a list of all value labels AFTER any plugins have been executed.
Long Data	contains all variables in long format after the reshape command has been executed.
Custom Data	contains all variables in long format that have been renamed with user defined customized variable names.
Project Retrieval	contains a list of stata data commands have been executed, allowing one to recreate the data set just created by SOEPMENU. If plugins were used, the "Collected Plugins" file is automatically called.
Collected Plugins	all add-ons that were executed on the data are collected here, which can be called by the "Project Retrieval" DO file.

Technical Appendix 5: Drop-Down Menus in SOEPMENU

The SOEPMENU system is a system of drop down menus added to the regular Stata drop down menus. You will find them under "User | SOEPMENU". This drop-down menu structure is defined by the Stata add-on `soep.ado`.

[Project]	Basic Project File Management
[New]	Deletes a project in memory; makes a "clean slate". Uses <code>smclearall.ado</code> .
[New, Open]	First makes a clean slate and then prompts the user to open a soep project using the Stata "open file" dialog box. Uses <code>smbaskopen.ado</code> .
[Save]	Prompts the user to save the project as a soep project file without prompting for a name. Uses <code>smbasksave.ado</code> .
[Save, As]	Prompts the user to save the project as a soep project file, while prompting for a name using the Stata "save file" dialog box. Uses <code>smbasksave.ado</code> .
[Prefs]	Defines the project preferences, such as: retrieved waves, region selection, sample selection, gender selection, path name to the SOEP data, plugins used, etc. Uses <code>smpref.ado</code> .
[Do the Retrieval Now!]	Executes a soep project in memory, thereby retrieving data and creating a wide and long data set. This only appears if a soep project has been opened. Uses <code>smproggen.ado</code> , <code>smproggenph.ado</code> and <code>carbcopy.ado</code> .
[Update]	Automatic updating of item labels. If the SOEPMENU project was created in year t-1, then the user needs to augment the SOEPMENU project with the missing year t variable information. This can be done automatically.
[(Step 1) Update Labels to]	The user selects the language for the updated items.
[(EN) English]	Updates item labels to English. The user must save the SOEPMENU project under a new file name. Uses <code>smupdatelangen.ado</code> .
[(DE) German]	Updates item labels to German. The user must save the SOEPMENU project under a new file name. Uses <code>smupdatelangde.ado</code> .
[(Step 2) Remaximize Plugins]	The SOEPMENU project ties in all possible plugins. Uses <code>smpluginadmin.ado</code> .
[(Step 3) Do UPDATED Project]	The updated SOEPMENU project needs to be rerun. The data at all levels are retrieved: raw, wide, long and custom. Uses <code>smproggen.ado</code> .
[Edit Existing Project Items]	Opens an item browse in SMCL format, allowing one to remove items already selected. Further, plugins can be manually added and deleted, automatically added and deleted, and simply viewed. Item view options are as follows: (Lo) simply a list of items, (Med) a list of items with the corresponding item correspondence and (Hi) a list of items with the corresponding item correspondence and also the variables used sorted by SOEP file. This is only selectable after a

	valid project has been open already. Uses <code>smviewfilebasket.ado</code> .
[Add New Project Items by ...]	Various methods to get more items into the project.
[Opening Data File]	Opens a tool which allows one to select SOEP data files from a drop down menu. Once the tool is open, one works from the top of the tool downwards. A SOEP file is selected and loaded. The "V" button means complete the particular stage one is at and drop down to the next level. Once a file has been loaded (like the stata command use), the items available are automatically scanned. These items then become available in the drop down menu system. The user can then select items and add them to the project. Once added however, the item can only be removed by selecting the menu item [Edit Existing Project Items]. Uses <code>smtool.ado</code> .
[Selecting Items from Data Browse Page]	Any SOEP related data file in memory is scanned for item-correspondence information. If this is found, it is made available for selection (mouse clicking).
[Appending another Project]	Assuming that a project has already been open, any other project can be appended to it, such that all items are added from the using project. Uses <code>smbaskappend.ado</code> .
[Global Lookfor Search]	Assuming that a project is in memory (already been "saved as" something), then the user may search all original SOEP data files sequentially, going backwards from the most recent year to the starting year 1984. All longitudinal files are opened and the labels scanned for the keywords given. Uses <code>smlookfor.ado</code> .
[Add New Project METAs by...]	
[Opening META Data File]	Use <code>smmetatool.ado</code>

(This area is only activated if a retrieval, e.g. "mysoepstuff" has been successfully run.)	
[mysoepstuff - Raw Data]	Things to do with raw data.
[Open]	Simply open the raw file created. Uses <code>smuse.ado</code> .
[mysoepstuff - Wide Data]	Things to do with wide data.
[Open]	Simply open the wide file created. Uses <code>smuse.ado</code> .
[Browse]	Opens a tool to browse wide data. Uses <code>smbrowse.ado</code> .
[Codebook]	Runs a variant of the stata command codebook. Uses <code>smlcbook.ado</code> .
[Dump out Label Codebook for Excel]	Dump out variable labels and value labels of wide variables as a tab separated ASCII file to be imported into an Excel spreadsheet. Uses <code>smclbook.ado</code> .
[Variable Order]	Change the physical order of variables found in the wide data set. This has nothing to do with the sort order!

[By Item]	For example: ap2277x bp2277x ... tp2277x, ap629x bp629x ... tp629x. Uses smsortwidevar.ado.
[By Wave]	For example: ap2277x ap629x, ..., tp2277x tp629x. Uses smsortwidevar.ado.
[mysoepstuff - Long Data]	Things to do with long data.
[Open]	Simply open the long file created. Uses smuse.ado.
[Browse]	Opens a tool to browse wide data. Uses smbrowselong.ado.
[Codebook]	Runs a variant of the stata command codebook. Uses smcodebook.ado.
[mysoepstuff - Custom Data]	Things to do with long data.
[Open]	Simply open the custom file created. Uses smuse.ado.
[View Run-Time]	Recreating the retrieval described in the soep project by scratch.
[Basic Data Extraction]	After the user has executed a soep project (run a retrieval), a Stata command file (DO) is automatically created, allowing the user to recreate the data set independent of the SOEPMENU system.
[Actual Plugins Used]	If any plugins were used or any other add-ons required, they are stored here. If a wide/long data set is recreated using the Run-Time files, then this file is called automatically by the main DO file described above.
[Required Citation in Paper]	This is the citation that you are required by contract to insert into all working papers, publications of any sort.

[Make Browse Page]	Creates a SMCL page for any data set in memory listing for ALL variables the variable name, variable label and any item correspondences found. If an item is found, it is made clickable, allowing the user to add it to the soep project by simply clicking on it. This is an alternative to the menu item [Opening Data File]. Also for each item listed in this page, clickable actions are also available: dstat, tab, kdensity, notes. Uses smviewfile.ado.
[Change Labels]	For SOEP data files and wide/long data files created from a soep project, the label language may be switched at will. Currently English and German are supported. Upon selection, all variable labels and value labels are redefined to be that of the language selected. This might take some time, if one has many variables or variables with many value labels such as industry or occupation indicators.
[Item Label Language]	Changes language of item labels in soep project.
[(EN) English]	Makes all labels English if available. Uses smitemlangen.ado.
[(DE) German]	Makes all labels German if available. Uses smitemlangde.ado.

[Var/Val Label Language]	Changes language of variable and value labels in a Stata data set.
[(EN) English]	Makes all labels English if available. Uses <code>smviewfileen.ado</code> .
[(DE) German]	Makes all labels German if available. Uses <code>smviewfilede.ado</code> .
[Values in Labels]	By default, the SOEP data files have value labels which contain the values themselves, e.g. 1 "[1] Man", 2 "[2] Woman". This can be stripped if desired, or added again. This works with any Stata data file. This might be desirable if you want to export data to SPSS, which displays values <u>and</u> value labels in frequencies anyway.
[Add]	Add "[x]" to all value labels, where "x" is the value. Uses <code>smnumlabel.ado</code> .
[Remove]	Remove "[x]" from all value labels, where "x" is the value. Uses <code>smnumlabel.ado</code> .
[Dump out Data for ...]	Any data in memory can be "exported" to another supported statistical package. For instance, having created a long file from a soep project, this file can be exported in to SPSS, SAS, Excel and SQL format. In all cases, the resulting data file is in ASCII format, and where appropriate a command file is also created to allow automatically creating a native version binary data file e.g. *.sav, *.sas7bdat, *.xls, etc. All variable labels and value labels are kept, except for Excel and Limdep which have no concept of value labels.
[SPSS (TM)]	Creates an ASCII data file and an SPSS command file to read in the ASCII data, and create a native format SPSS file. Uses <code>smsta2spss.ado</code> .
[SAS (TM)]	Creates an ASCII data file and an SAS command file to read in the ASCII data, and create a native format SAS file. Uses <code>smsta2sas.ado</code> .
[LIMDEP (TM)]	Creates an ASCII data file and a LIMDEP command file to read in the ASCII data, and create a native format LIMDEP file. Uses <code>smsta2limdep.ado</code> .
[R-Project (TM)]	Informs the user that R can import Stata data sets directly. Uses <code>smrproj.ado</code> .
[EViews (TM)]	Informs the user that EViews can import Stata data sets directly. Uses <code>smeviews.ado</code> .
[Excel (TM)]	Creates a tab-delimited ASCII data file for importing into Excel. Uses <code>smspreadsheet.ado</code> .
[MySQL (TM)]	Creates an ASCII file with MySQL Commands allowing the automatic creating of an MySQL data base. Uses <code>smtosql.ado</code> and <code>smsqlbatch.ado</code> .
[Initialize Database]	Using MySQL commands, a database named "soep" is defined in the file <code>soep.sql</code> , comprising of empty tables for variable labels "varlab" and value labels "vallab". Uses <code>smtosql.ado</code> .
[Add Data in Memory to Database]	Using MySQL commands, any data in memory is written as "inject" statements (for increased portability) to a file named <code>datafile.sql</code> . Uses <code>smtosql.ado</code> .
[Add Entire SOEP Directory to Database]	Using MySQL commands, it does just that. It takes a long time (easily 18-24 hours), however it is about as portable as SQL gets.

Variable labels and value labels are saved in the SQL database.
Uses `smsqlbatch.ado`.

[**Help**] Starts your default browser and points to various web pages or starts an Email program.

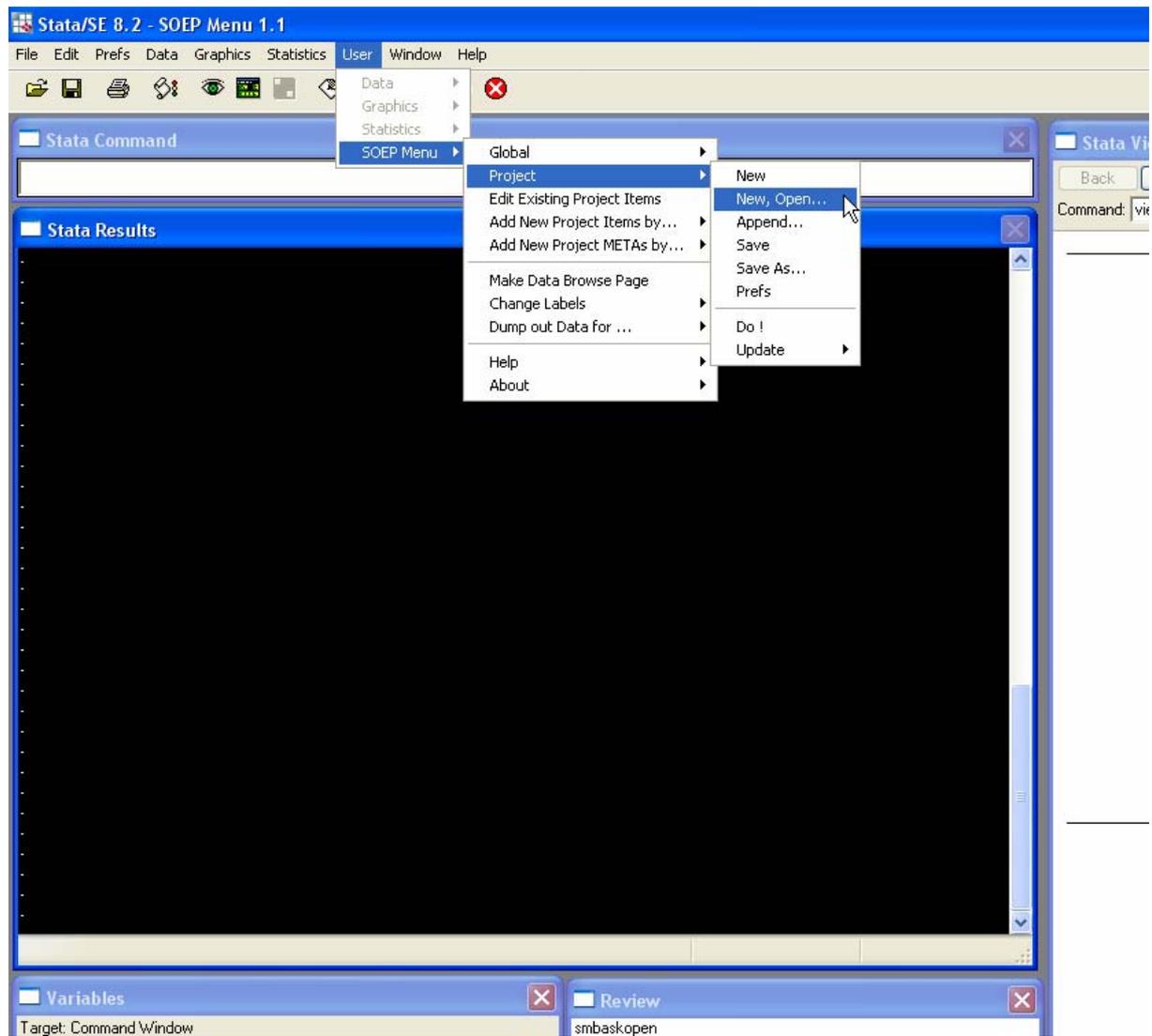
[SOEPMENU]	Various SOEPMENU help pages.
[What's New?]	Read a Stata SMCL text file in which known bugs and bug fixes have been listed. Reads the file <code>whatsnew.smcl</code> stored in the <code>base</code> directory.
[Official Updates]	Check online to see if an updated version is available for download from the SOEPMENU webpage. Uses <code>smupdatechecksm.ado</code> .
[SOEP Menu Web Site]	Starts your default browser and points to http://www.SOEPMENU.de .
[Pre-Programmed F-Keys]	Lists the available F-Key shortcuts.
[Full Documentation as PDF]	This file opened as an Adobe Acrobat PDF file
[Email Author]	Starts default E-mailer to send an email to the author.
[SOEP Web Site]	Starts your default browser and points to the SOEP Website at the DIW Berlin (http://www.diw.de/soep).
[Main Web Site]	The Main SOEP web page, with all kinds of documentation etc.
[SOEPinfo]	The cool web tool that Ingo Sieber and I developed over the period 1996-2002, and where he continues on! SOEPinfo is similar in functionality to SOEPMENU (http://panel.gsoep.de/soepinfo).
[SOEPLIT]	A web based tool to search the SOEP's database of SOEP literature. Don't forget, if you write a paper, report, article, book, dissertation, thesis, etc using the SOEP, you are required by your SOEP contract to give a copy to the SOEP group as soon as possible! (http://panel.gsoep.de/soeplit).
[DIW Berlin]	The home of the SOEP in Berlin. (http://www.diw.de).
[SOEP User Support]	Write an email to the SOEP user support hotline.
[SOEPinfo Support]	Write an email to Ingo Sieber to get help on SOEPinfo.
[World Panel Data]	Various links to homepages of panel data providers. The data providers are separated by geographical region. SOEPMENU only works with the SOEP.
[About]	Some other smaller bits of information
[Warning]	You alone are responsible for what you do with SOEPMENU.
[Legal Info]	Please don't misuse SOEPMENU.
[Thanks]	Some people I wish to thank.
[Please support UNICEF]	Points your web browser to the International UNICEF Website, so that you can find information on how to make a donation.
[UNICEF Deutschland]	Points your web browser to the German UNICEF Website, so that you can find information on how to make a donation.

Technical Appendix 6: Retrieval Options

[Unit of Analysis]	Basis for retrieval: Person or Household
[Person]	Default. Creates one observation per person in a wide file and person-year observations in a long file.
[Household]	Creates one observation per household in a wide file and household-year observations in a long file.
[Panel Design]	This determines who is in the output data. Not everyone in the SOEP takes part every year.
[Unbalanced Panel]	Default. Persons taking part in ANY year are added to the wide file. In the wide file, only those person-year (or household-year) observations are kept where the person (household) actually took part.
[Balanced Panel]	Persons (Households) MUST taking part in ALL years.
[Years]	Year span for the retrieval.
[All Years]	All available years are used.
[Post Unification]	Only 1990-200x are used.
[Pre Unification]	Only 1984-1989 are used.
[xxxx-yyyy]	The user must type in himself the year span, e.g. 2000-2003. If only ONE year is required, the year span is typed in as 2000-2000.
[Gender]	Immediate selection on a specific gender. Relevant only for person based retrievals.
[All]	Default. Select both males and females.
[Males]	Select only males.
[Females]	Select only females.
[Region]	Immediate selection on a specific geographical region (west or east).
[All]	Default. Select both West Germany and East Germany.
[Just West]	Select only residents of West Germany.
[West+Changers]	Select only residents of and movers to West Germany.
[Just East]	Select only residents of East Germany.
[East+Changers]	Select only residents of and movers to East Germany.

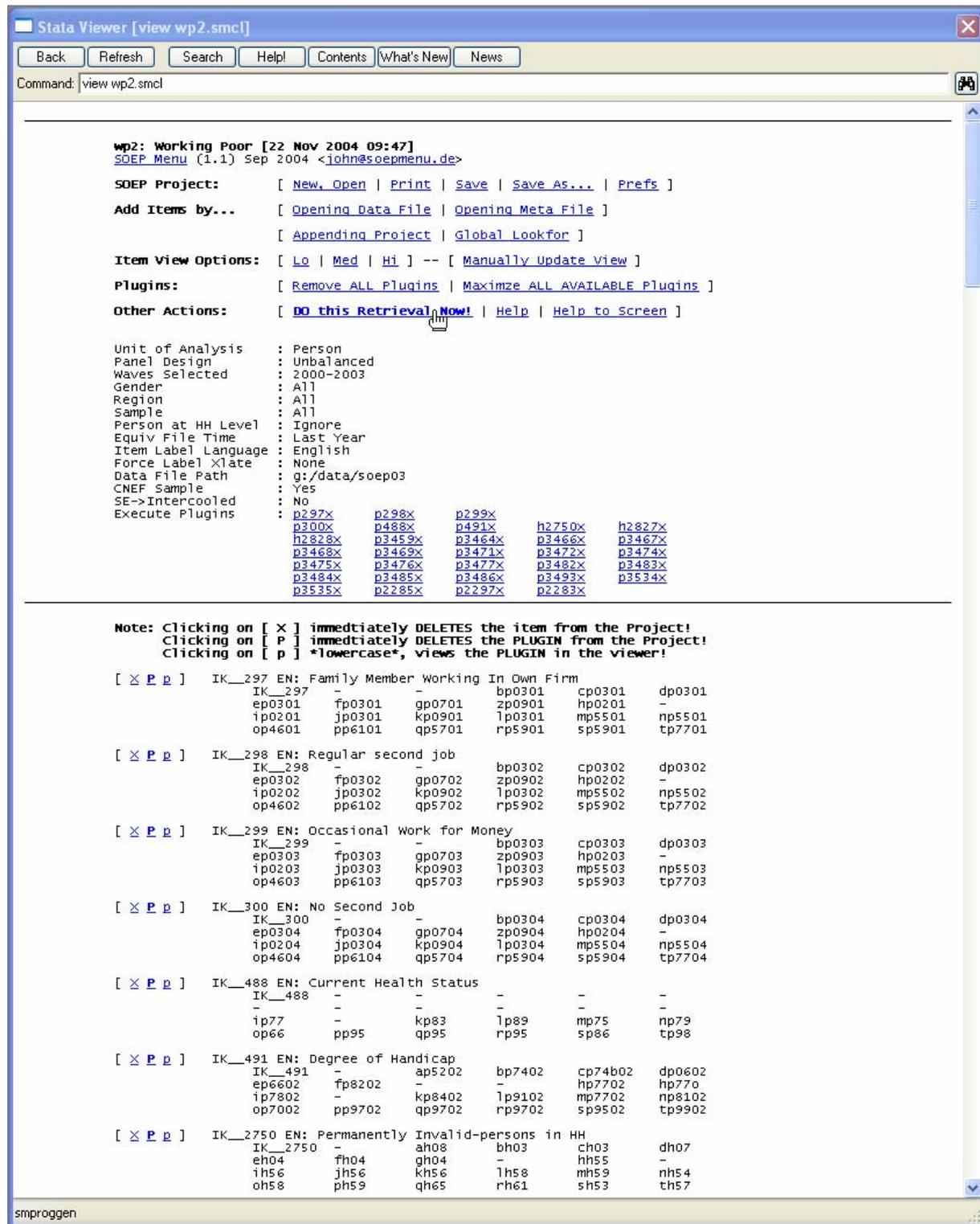
[Sample]	Immediate selection on a specific SOEP subsample. This is an advanced feature.
[All]	Default. Select both West Germany and East Germany.
[Sample A-E]	Select only subsamples A-E.
[Sample F]	Select only subsample F.
[P in HH Ret]	How to deal with person-specific information in a household retrieval. As most households contain many individuals, it is then unclear which person's information is to be included. This is an advanced feature.
[Ignore]	Default. Person-specific variables are not retrieved when the unit of analysis is the household.
[Head]	Person-specific variables are first matched to the values given by the household head when the unit of analysis is the household.
[Spouse]	Person-specific variables are first matched to the values given by the spouse of the household head when the unit of analysis is the household.
[P in HH Ret]	How to deal with person-specific information in a household retrieval. As most households contain many individuals, it is then unclear which person's information is to be included. This is an advanced feature.
[Ignore]	Default. Person-specific variables are not retrieved when the unit of analysis is the household.
[PEQUIV Income]	Detailed income information in the PEQUIV files (the German component of the CNEF files) are asked in year t about year t-1.
[Last Year]	Default. Variables are saved as normal.
[This Year]	Variables with names starting in "i" are saved as if they came from the current year.
[Plugin Info]	Detailed information contained in the Plugin files is displayed on the screen as the plugins are being run.
[No]	Default. Minimal Information.
[Yes]	Detailed information displayed.
[CNEF Sample]	Retrieval settings are set to be optimal for CNEF. This implies that males, females and children in both east and west are extracted in the data. CNEF "i" variables are saved as "last year".
[No]	Default.
[Yes]	Retrieval settings optimized for CNEF.

Figure 1: Screen Dump of Menu System



Note: If the Stata profile has been set up correctly, SOEPMENU will automatically load each time Stata is started. See Technical Appendix 1 for details.

Figure 2: Screen Dump of Example SOEP Project File



Note: This is the SOEPMENU project file containing many items, e.g. p297x, p298x etc. The underlying SOEP variables are listed below each item.

Figure 3: Screen Dump of SOEPMENU Project Preferences

SOEP Preferences

Unit of Analysis	Panel Design	Years
Person	nbalanced	2000-2003
Gender	Region	Sample
All	All	All
P in HH Ret	PEQUIV Income	SE -> Intercooled
Ignore	Last Year	No
Item Lang	Force Xlate	Plugin Info
English	None	No
CNEF Sample		
Yes		

Item Plugins: na na p297x p298x p299x p300

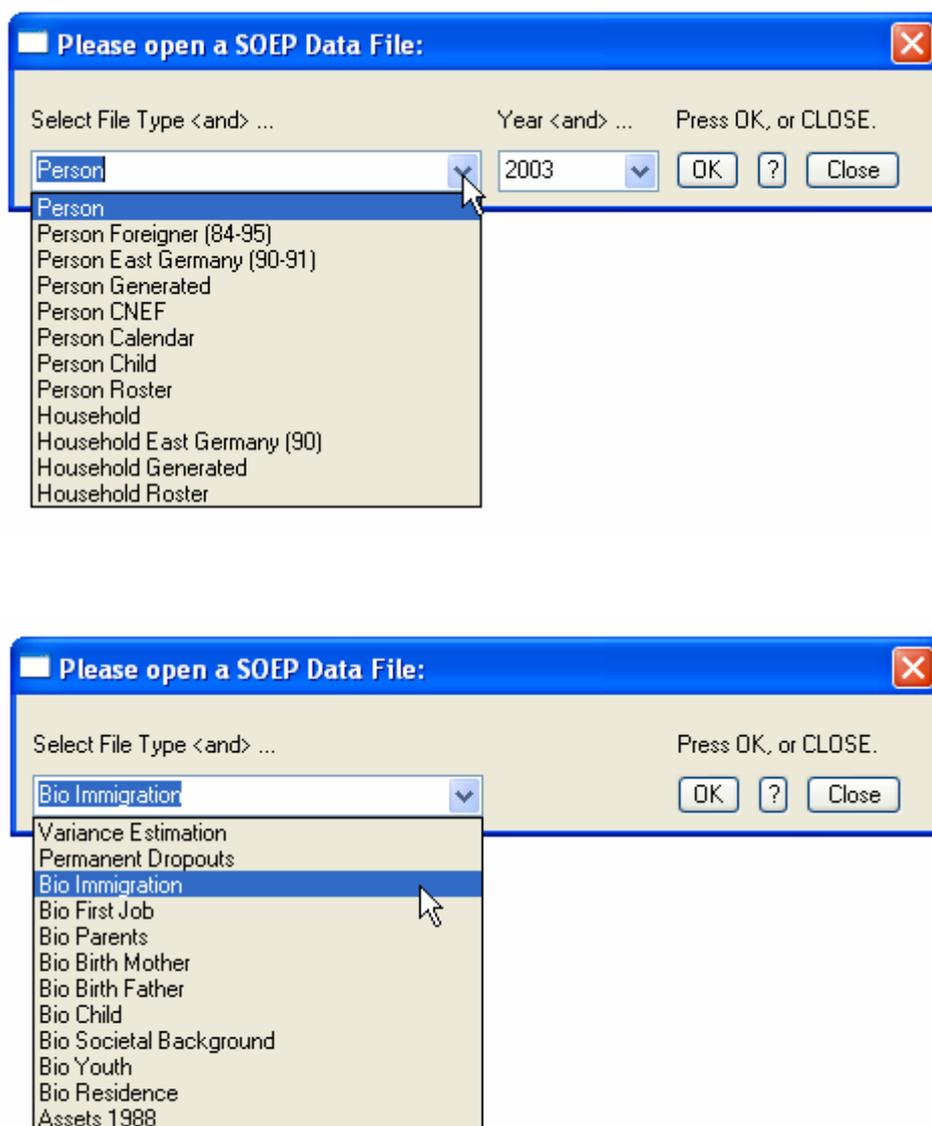
Path to SOEP: g:/data/soep03

Comment: Working Poor

Apply Close

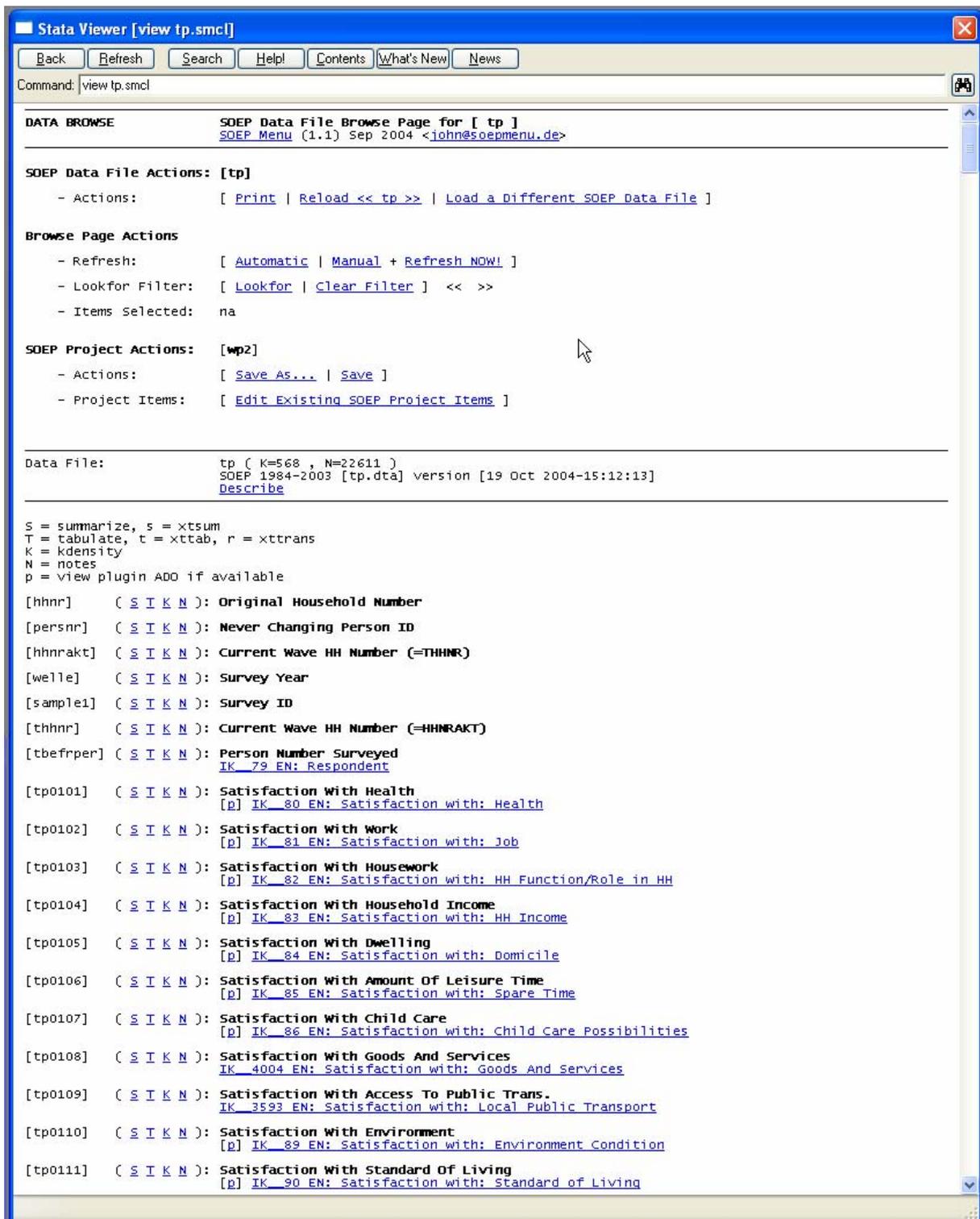
Note: Here the retrieval preferences are determined for a specific SOEPMENU project.

Figure 4: Screen Dump of Possible SOEP Data Files to Open



Note: Here SOEP data files can be opened: (a) above the regular wave specific files, e.g. person and household and (b) below Meta files without the usual time dimension. Please note that Spell-data files have not yet been implemented into SOEPMENU.

Figure 5: Screen Dump of SMCL file created upon opening a SOEP Data File



Note: Each variable in the data file is listed with the variable label and underneath the item labels that might be contained in the variable. Item labels can be selected by clicking on them. Item labels preceded by a [p] indicate that a plugin is available for the item.