Heike, Hans-Dieter; Beckmann, Kai; Fleck, Claudia; Ritz, Harald



# The Darmstadt micro-macro-simulator: Consistency check and data modelling of GSOEP

Vierteljahrshefte zur Wirtschaftsforschung

**Provided in Cooperation with:**
German Institute for Economic Research (DIW Berlin)

*Suggested Citation:* Heike, Hans-Dieter; Beckmann, Kai; Fleck, Claudia; Ritz, Harald (1994) : The Darmstadt micro-macro-simulator: Consistency check and data modelling of GSOEP, Vierteljahrshefte zur Wirtschaftsforschung, ISSN 0340-1707, Duncker & Humblot, Berlin, Vol. 63, Iss. 1/2, pp. 139-144

This Version is available at:
https://hdl.handle.net/10419/141062

## The Darmstadt Micro-Macro-Simulator — Consistency Check and Data Modelling of GSOEP

by Hans-Dieter Heike, Kai Beckmann, Claudia Fleck and Harald Ritz

Microsimulators transfer a representative sample of decision units from period t to period t+1. They are developed to improve the explanatory power and forecasting capabilities of economic theories and are used to simulate economic and sociopolitical programs. However, simulations across several time periods can be accomplished only if a cross-section microsimulator is linked to a macromodel processing time series data. In this way, time series paths can be taken into account.

The Darmstadt macrosimulator is not of the equilibrium type but allow disequilibrium caused by non-market clearing prices and inconsistency in hypothetical supply and demand plans. We are able to simultaneously link a macro- and micromodel within our object-oriented version of the DMMS. Convergence was reached generally after 10 — 20 iteration loops.

The micro database of our DPMS has been changed from the EVS (Income and Expenditure Survey of the Federal Statistical Office) to the SOEP because of the obvious advantages of the SOEP (probability sample, covering all population groups, information on numerous living areas, computation of stocks and flows, gross and net effects are possible, causal analysis and investigation of the effects of lagged variables are enabled). But the SOEP also has drawbacks (panel effect, panel mortality, sensitivity against data errors).

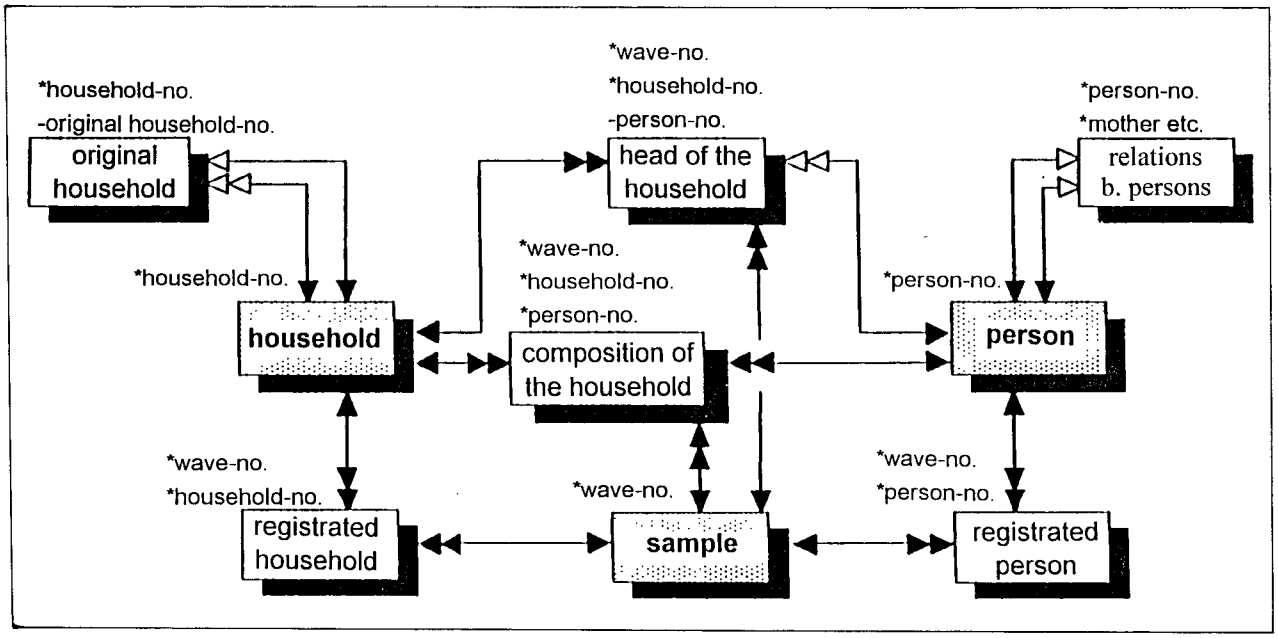
To support the use of the SOEP as the database for our DPMS several SOEP data models have been developed. Our aim was to get a data model that provides a consistent database and efficient selection paths, also serves as a template for importing the SOEP data into the relational database system. This paper discusses the ER-Modelling of the SOEP. Our work was the basis for improving the quality of data in the GSOEP.

### First Modelling Steps

All modelling proposals were performed according to the ISOTEC standards of the Ploenzke AG. The basis of the Information Structure Analysis (ISA) is the Entity-Relationship-Model (ERM) first published in 1976 by Peter Chen. Specifically these data models should include uniform access paths for cross-section and time series data, eliminate inconsistencies, avoid redundancies and pay attention to the large number of attributes of each person or being interviewed (about 1500) and to the modelling of the time component of time series data. We started with the following ground model, completely abstracting from the factual structure of the data (see Figure 1):

All persons and households registered in at least one wave are represented in the data model independent of the individual sampling. Households are marked by household numbers (key attribute), persons uniquely by personal numbers. The temporary aspect is referred to by the information object "sample". Each sample is labelled by a wave number. The wave-data of each registered household or person are now represented by the information object "sampled household" or "sampled person" respectively. These actually sampled objects (household and persons)

Figure 1



are related to the registered objects and to the individual wave within which the data are collected.

Relations between households and persons are modelled by means of the information objects "household structure" and "chief of the household". Since the household structure is changing in the course of time, these are 3-place predicates that relate a household, together with a person, to a distinct wave. Furthermore relations between households and relations between persons have to be taken into account. A concrete example is the relation of a household to its stem household, that is, to the household from which it has branched. On the basis of this first data model, the actual SOEP data model has been developed considering the structure of the raw data.

### Realized Alternative

The structure of the yearly raw data first had to be analysed on the basis of the available documentation in order to identify the information objects and their relations. They then had to be described formally. The relevant object-types and their relations have been identified by the SOEP variable catalogue, the SOEP user manual, the questionnaire of each wave and the item-correspondence-list (ICL).

Finally an ERM with 80 entities has been formulated which is subdivided into seven partial models for reduction of complexity. Partial model 1 represents the fundamental model. Within this partial model the household- and person-related objects and their relations are modelled on the basis of the object "sample wave (EHB)". A differentiation of the object "net household (NHH)" is then performed in partial model 2.

As expected, a large number of describing attributes were associated with the object "adult (EWA)". Mode of questioning, interview method, demographical attributes, pension and taxes are related to this object. The very large number of relations to other objects implies that the entity "adult (EWA)" is the central entity of partial model 3. Figure 2 shows this model. The wave-dependence of the objects is documented by the primary keys "wave-no" and "person-no".

The four other partial models are refinements of the object "adult (EWA)". On one hand it is split into the objects "German (DEU)" and "foreigner (ALD)" through a complete disjoint specialization. On the other hand the object "adult (EWA)" is subdivided by a complete and disjoint specialization into the object "unemployed person (NEA)" and "employee (ERW)".

### Consistency Check of the SOEP

Early on we found that the code-meaning of an attribute differed between waves and inconsistencies appeared within the keys of the code-meaning. We decided therefore to perform a program-supported consistency check of the available first six panel waves at our institute.

The dialogue program we developed had to show sequentially all attributes and their response categories through all waves. Having achieved this, we could decide whether deviations of answers and their key assignments (encryption system) were caused by a syntactic or even by a semantic difference and try to eliminate inconsistencies. A classification of these inconsistencies is presented below subdivided accordingly to their appearance in cross-section or longitudinal analysis and whether elimination is possible or not. Furthermore errors in DIC-files and in the ICL are mentioned. These inconsistencies are partially eliminated in the newer versions of the DIC- and DAT-files by way of an exchange of information between our institute and the DIW.

Classification and elimination of inconsistencies was made possible by systematic electronic data processing methods that comprise modern software engineering methods of data modelling as well as program supported import of SOEP-data into a relational database system.

Before we present examples of localized inconsistencies and describe the quantitative results of our inconsistency checks, we present some details of the program-technical implementation of the consistency check.

## Software-Technical Realization of the Consistency Check

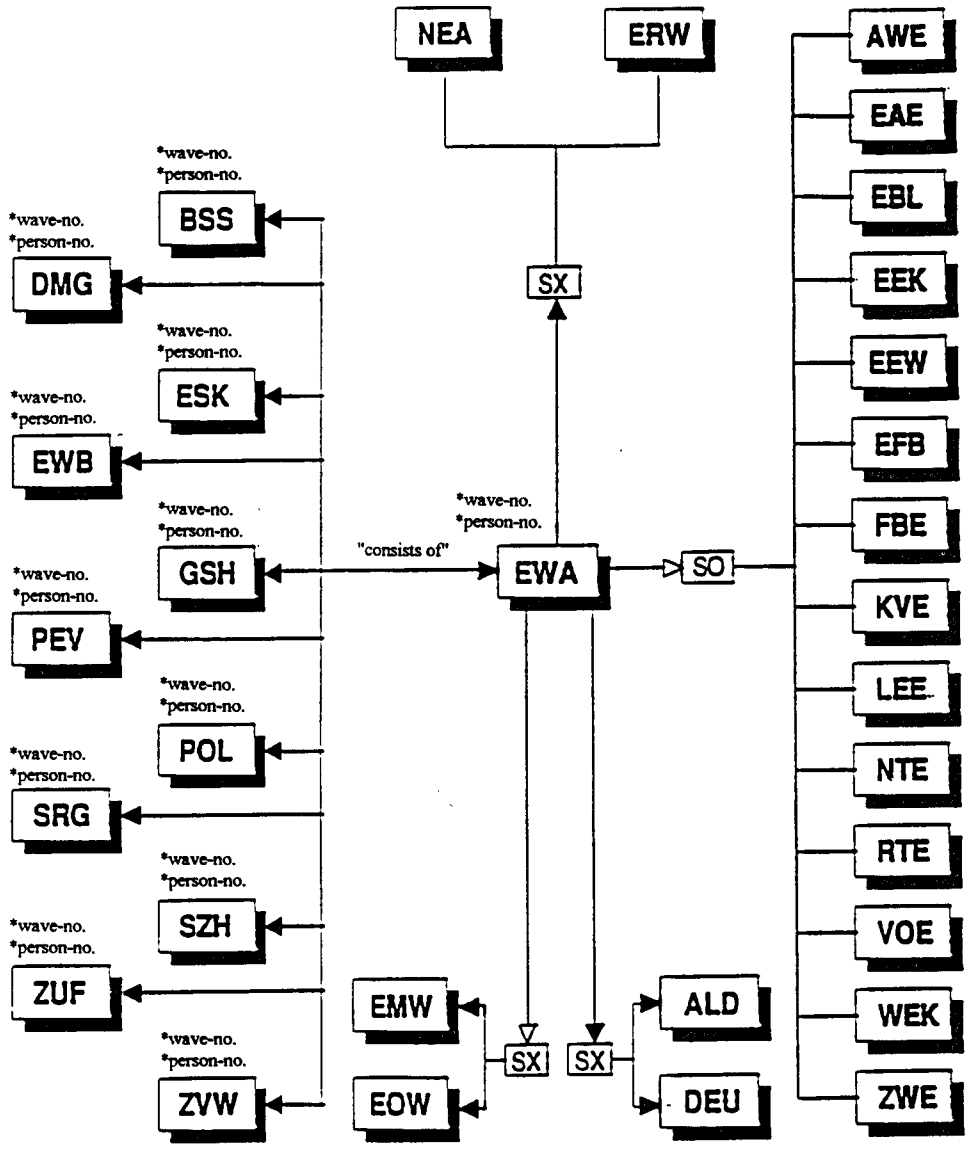
Consistency checks were carried out on the basis of the DIC-files of the first six panel waves and of the ICL that the DIW provided some years ago. Our consistency checks established a modified ICL, the so-called "own-ICL" that includes nothing but the information on consistent data. The variable-coding for "own-ICL" always provided the best coding of the variable in question, because often in subsequent waves the range of possible answers was widened. The "own-ICL" has an important influence on the implementation of the import interface that allows the transfer of panel data into a relational database. Turbo-Pascal 6.0 was used on IBM-PS/2 systems at our institute.

Our first Pascal program performed an automatic consistency check of the panel DIC-files of all six waves and the DIW-ICL. Variables with inconsistencies (mostly wave-dependent variations of code-values and code-meanings) were registered in an error file; all other variables were taken out in the "own-ICL".

The error file was handled by a second Pascal program. Consistent variables were transferred into the "own-ICL" thereafter. The "own-ICL" generated by the Pascal programs includes the description of all variables transferred into the relational database. A third Pascal program attributed a unique name and an object of the data model with each variable.

The ADABAS database is generated by the fourth Pascal program. It generates, using the "own-ICL", a JCL batch program which contains commands necessary to generate the relational ADABAS database at the IBM 3090-200E

Figure 2




ALD foreigner
AWE adult in an education or further education program
BSS valuation of social security
DEU German
DMG demography
EAE self employed adult
EBL adult with examination in the last year
EEK adult with income and transfers
EEW adult with first employment
EFB adult with family biography
EMW adult with further education
EOW adult without further education
ERW employee
ESK status of employee
EWA adult
EWB employee biography
FBE adult with driving licence
GSH health
KVE adult with health insurance
LEE adult with wage and income taxes
NEA unemployed person
NTE adult with additional income
PEV personal income and property
POL political attitude
RTE retired person with pension
SRG sorrows
SZH social origin
VOE adult with membership in an association
WEK female adult with children
ZUF contentment
ZVW time consumption
ZWE adult with second dwelling

mainframe under VMS/ESA. With the help of a fifth and sixth Pascal program final adjustments for the import interface on the mainframe are made.

### Cross-sectional Inconsistencies

Cross-sectional inconsistencies, i.e., inconsistencies concerning exactly one wave, are caused by errors in the codification of variables documented in the DIC-files. A false codification of a variable means that different code-meanings have the same code-value. For instance, in the codification of the variable "nationality" the code-meanings "Columbia" and "Ethiopia" both have the code-value (47). Unless such an error can be corrected the variable cannot be integrated within a consistent database. In most cases our data check revealed that the cause for these errors was just typing errors, so that the variable was not lost.

### Longitudinal Inconsistencies

Comparison of the first six panel waves showed a variety of inconsistencies, which can be classified as follows: *(1) Extended code-meaning of a code-value* caused by an incomplete translation of the questionnaire into the DIC-files; *(2) Different syntactical code-meanings* caused by using different abbreviations for the same word or by using different expressions for the description of a code-meaning; *(3) Different code-meanings of a code-value* caused by a change of concept used within possible answers of a question in the questionnaire or by a simple change in the order of the same possible answers of a question; *(4) Different code-values for the same code-meaning* mainly caused by inconsistent codification of continuously extended answering possibilities, typing errors, or a change in the order of the same possible answers of a question.

Inconsistencies of error-types 1 and 2 can easily be solved by examining the SOEP-questionnaires, so that there is no wave-specific loss of data; whereas error-types 3 and 4 usually would lead to a wave-specific loss of data unless one creates a "new" complementary consistent variable that refers to the wave-specific inconsistencies.

### Sources of Error

*Errors of the DIC-files:* Our study revealed a number of variables registered under "data list" in the DIC-files that did not get a description of content under "var labels" or, in case of a coded variable, the codification under "value labels" is missing or even wrong in its contents. For some variables neither can be found. For some variables listed under "var labels" the descriptions of content do not conform longitudinally, and in the worst case another description of content can be found in the item-correspondence-list. Errors of this kind may be eliminated by means of the documented questionnaires in the SOEP user manual. Otherwise the data would be lost.

*Errors of the ICL:* A variety of errors and shortcomings were also found in the item-correspondence-list (ICL) . Some variables got wrong wave-specific variable names, mainly caused by typing errors. These errors were easily corrected by means of the DIC-files. The ICL contains 45 wave-specific variable names that cannot be found in the DIC-files. This could also be an error in the DIC-files. Because such errors usually cannot be traced by the user, these variables are regarded as non-existent.

### Quantitative Results of the Consistency Check

A first evaluation of the results of the consistency check shows that only 492 of the 643 of the wave-specific variables referenced in the DIW-ICL of the first wave remain in the "own-ICL". Over all six waves, 4430 variables out of 5257 referenced in the DIW-ICL remain in our "own-ICL". Of the initial 1643 variables (from now on called "attributes"), 1313 remain after the consistency check. The difference is caused by inconsistencies and by variables that could not be traced.

The „own-ICL" could be evaluated in accordance with the criterion „availability of the single attributes in each wave". 541 attributes (about 40 percent of the transferred attributes) are available in only one wave. Only 375 attributes (30 percent) are available in all waves. This shows the degree of inconsistency in the data despite all efforts to eliminate those shortcomings.

## SOEP Data Administration in ADABAS

Having performed the consistency check of the data and generated the ERM of the SOEP data, the next steps are to generate the ADABAS database using the ERM and to develop and implement a selection program using our consistent database.

### Software Technical Realization of the SOEP Import Interface

The "own-ICL" and the key file are transmitted into the ADABAS files "EIGEN-IKL" and "SCHLUESSELDATEI". Through the NATURAL program "IKLEIN-P" respectively "DATEIN-P". The NATURAL-program "HAUPT2-P" is then transmitting the attribute values from the wave overlapping DAT-file DATHHRF, DATHPFAD, DATPHRF and DATPPFAD into the relational database.

The NATURAL-program "ATTRIB-P" transfers attribute numbers and their denominations from the "own-ICL" into the object „SOEP-ATTRIBUTE". "KODIER-P" moves the code-values and their meanings out of file "ATTRKWKB.TXT" to the object "SOEP-KODIERUNGEN" within the relational database.

The import of data out of the DAT-files that are directly associated with waves into the different objects of the

database is carried through by the NATURAL program "HAUPT1-P". Necessary parameters are delivered to "HAUPT1-P" by a batch program written in Job Control Language (JCL); this program also starts "HAUPT1-P".

Program "HAUPT1-P" transfers a DAT-file line by line and stores the key and attribute values contained within the relational database by mean of subprograms. The key and attribute values are identified with the assistance of the key file and the "own-ICL".

Attached to each imported object is a subprogram which performs the write commands into the database. It became necessary to specify the entire set of attributes of all objects since the denomination of the attributes of an object had to be formulated explicitly.

Importing the panel data into the ADABAS database under VMS/ESA on the IBM-mainframe 3090-200E at the TH Darmstadt 240 MB of space.

## SOEP Information Database

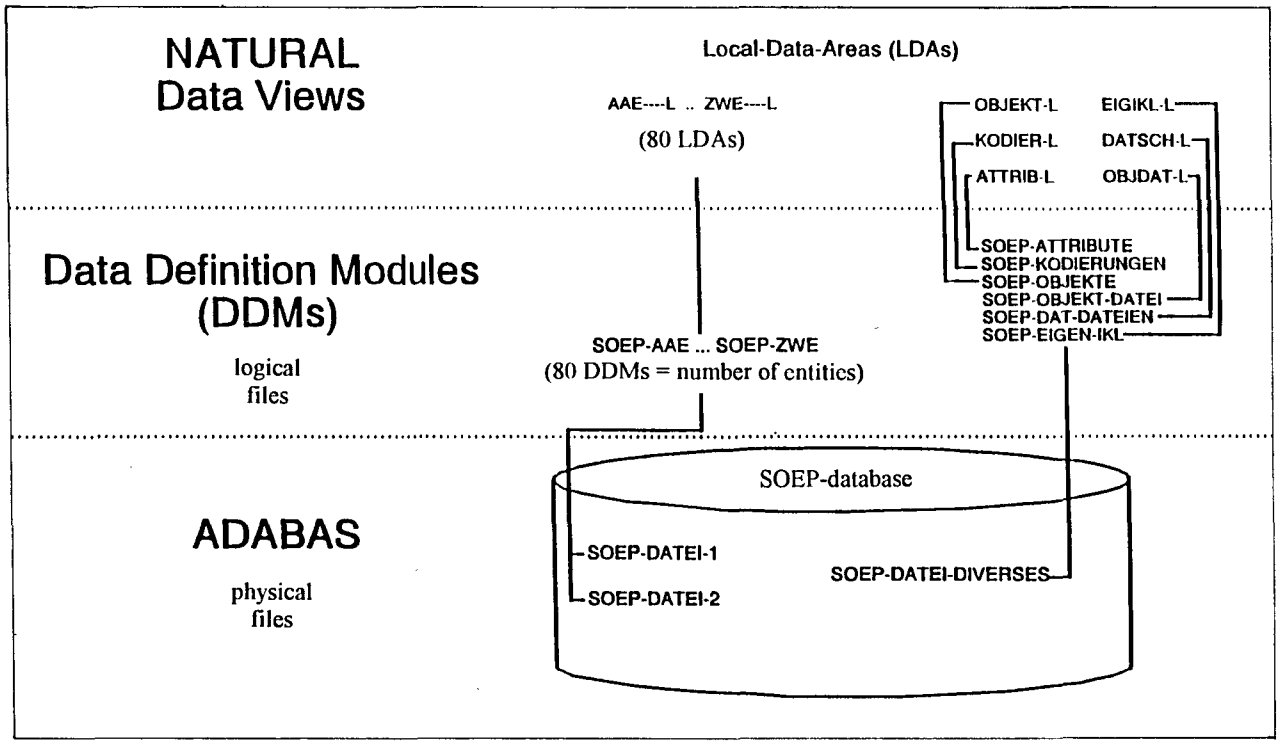
In addition to raw data, our ADABAS SOEP database contains a description of these data. This "metadata" supports applications of the imported data. We call this established database an information database. Figure 3 shows the design of this SOEP information database schematically. Within this database there exist three physical ADABAS files. "SOEP-Datei-1" and "SOEP-Datei-2" include the raw SOEP data.

The third file within the SOEP database is the "SOEP-DATEI-DIVERSES" which contains the metadata. The "SOEP ATTRIBUTE" file contains information on each attribute and each wave. The information includes attribute numbers and denominations as well as indications of whether data of a specific attribute within a specific wave is available or not. This information is intended to support applications of the imported panel data.

The file "SOEP-KODIERUNGEN" includes code-values and meanings. The logical file "SOEP-OBJECTS" contains information on each object of the data model concerning how many and which attributes and key attributes are related to each object. It is assumed that data are transmitted into at least one attribute of the object in question, that is, at least one attribute of the object is included within the "own-ICL".

The file "SOEP-OBJEKT-DATEI" tells how many attributes of object data out of a distinct DAT-file are transferred. This information appeared to be very useful in importing objects with the key attribute "household number", since these data could be sorted easily according to object name or DAT-file name by means of two superdescriptors. The file "SOEP-DAT-DATEIEN" includes the file name and wave of all DAT-files as well as "offsets" and formats of the key fields of person and household number in a DAT-file. The last file of "SOEP-DATEI-DIVERSES" is the "SOEP-EIGEN-IKL". This file includes the "own-ICL" at PC level and therefore represents the link between the DAT-file and the relational databases.

Figure 3



### SOEP Selection Program

The selection program allows user-friendly automatic access to the SOEP data by means of a SAA-user interface realized at a NATURAL/ADABAS development environment. The user will be supported for database requests related to both cross-sectional and longitudinal applications. Selections of importance to the user may be stored within an archive and can be activated at any time.

Moreover, a prototype selection program has been established in Pascal that performs selections on the raw database stored in the Novell-server of our institute. Program usage is facilitated by a mouse-supported SAA interface. A specific selection of attributes of the panel population or of a subpopulation can be performed. Access to the panel data is restricted by the "own-ICL" established in connection with the consistency check.

When specifying data selection the user may delete unwanted program options. A variable list contains all available variables and their variable numbers. Data examples are supplied as well as code-values and code-meaning. A specific variable list can be produced that includes only the information that is of interest to the user. By restriction of variables the user can define a subpopulation of the cross-section population, for instance, by restricting the value range of single variables. No more than 20 variables may be restricted within a single selection. The user then determines variables and waves of the data output relating to the specified subpopulation. Finally, three formats of the output file are offered in order to enable uncomplicated further processing of the data by means of standard software. An output file may be transferred without problems to a spreadsheet like EXCEL.

### Outlook

Our work was the basis for an improved version of the GSOEP data that is now being disseminated by the DIW. Further work on the DMMS relates to optimizing the access to such a large database by automatic design of the physical database structure which is optimized iteratively. The first set of results are promising. This research field is of special interest to computer scientists who work with distributed databases.

**References**


*Heike, H.-D.* and *A. Kaufmann,* 1987, Charakterisierung und Vergleich des Sfb3 und des Darmstädter Mikrosimulators. In Angewandte Informatik 4: S. 9-17.

*Heike, H.-D., O. Hellwig,* and *A. Kaufmann,* 1988, Der Darmstädter Pseudomikrosimulator, Modellansatz und Realsierung. In Angewandte Informatik 1, S. 9-17.

*Heike, H.-D., O. Hellwig,* and *A. Kaufmann,* 1988a, Das Darmstädter Mikrosimulationsmodell — Überblick und erste Ergebnisse. In Allgemeines Statistisches Archiv 72(2): S. 109-129.

*Kaufmann, A.,* 1988, Systematische Entwicklung von Mikrosimulationssoftware. Dissertation, Technical University Darmstadt.

*Heike, H.-D., K. Beckmann, A. Kaufmann,* and *Th. Sauerbier,* 1993, Der Darmstädter Mikro-Makro-Simulator — Modellierung, Software Architektur und Optimierung. Paper presented at the "7. Konferenz über die wissenschaftliche Anwendung von Statistik-Software (SoftStat'93)", Heidelberg, Germany, March 15-18, 1993. Monograph forthcoming.