Working Paper
ENGLISH ONLY

UNITED NATIONS ECONOMIC COMMISSION FOR EUROPE (UNECE)

CONFERENCE OF EUROPEAN STATISTICIANS

EUROPEAN COMMISSION

STATISTICAL OFFICE OF THE EUROPEAN UNION (EUROSTAT)

Joint UNECE/Eurostat work session on statistical data confidentiality
(Helsinki, Finland, 5 to 7 October 2015)

Topic (v): Practicum: Case Studies and Software

A Graphical User Interface to Manage Cell Suppression on Sets of Linked Tables Using SAS and τ-Argus

Sarah Giessing[*], Sven Grunwald[**]
[*] Statistisches Bundesamt, 65180 Wiesbaden, Germany, Sarah.Giessing@destatis.de
[**] Statistisches Bundesamt, 65180 Wiesbaden, Germany, Sven.Grunwald@destatis.de

Abstract: In several statistics of the German Statistical System we use the efficient τ-ARGUS Modular algorithm for secondary cell suppression. All those statistics produce large sets of multiple linked tables. In principle τ-ARGUS Modular can handle sets of linked tables in a single application. However, this requires that certain conditions concerning the structure of those tables hold. In practice we often face the problem that the table model needed to manage the disclosure risks of data foreseen to be published is very complex and does not meet those requirements. The protection process involves then multiple applications of τ-ARGUS where the outcome of one step affects the input for the following ones. In order to handle this efficiently we use a procedure implemented as set of SAS macros (Schmidt and Giessing, 2011). Metadata for procedure control are stored in several Excel sheets. We are now developing a graphical interface for this tool. Although this is still work in process, the paper presents screen shots of early versions.

1 Introduction

As argued for instance in (Giessing, 2013) the modular optimization algorithm (Wolf, 2002) offered by the software package τ-ARGUS (Wolf et al., 2014) is a very efficient algorithm for secondary cell suppression. This paper is in the context of integrating ARGUS into a production environment. Basically the τ-ARGUS software concept foresees simple flat files (e.g. txt, csv, ascii formats[1]) for data input and output. Of course, within a statistical production process data will usually be stored in other, typically data base formats. The idea is that users of τ-ARGUS extract the data needed for the confidentiality process from the data base, turning it to the simple format

[1] Recent versions of the package also support SPSS file formats.

readable by ARGUS. However, for regular, automated applications from within a production chain a better integration into the process is desirable.

One example for such an approach is the Bifrost system of Statistics Sweden (Almberg et al., 2013) consisting of τ-ARGUS (in an installation along with the commercial optimization solver package Xpress) and SAS2Argus, a collection of SAS macros that facilitate the use of τ-ARGUS via SAS (Kraftling, 2011). SAS2Argus basically follows the concept of the τ-ARGUS batch command file (c.f. Wolf et al., 2014, 5.7). However, SAS2Argus users will not directly use the Argus batch-file command syntax, but will supply the "parameters" of an application as SAS macro variables. When executing SAS2Argus, the macros will automatically generate data and metadata files in the ARGUS format, along with a τ-ARGUS batch command file specifying the application. They automatically execute τ-ARGUS in this configuration, read the τ-ARGUS output files and turn the output into the format specified by the respective SAS2Argus macro parameter.

At Destatis, we have developed very similar SAS routines, but these are just one building block of a much larger and complex package (Schmidt and Giessing, 2010 and 2011). A major intention of the package is to handle an SDC process involving multiple, linked tables as single application, also in a situation where table structures are too complex for a single application of τ-ARGUS. In our experience, this is rather the rule than the exception. The package serves then as control centre, executing multiple ARGUS applications, where the outcome of one application must be reflected in the input for the next. As in our IT environment it is more efficient this way, we have also developed another SAS tool for the aggregation process replacing this part of the ARGUS functionality. When many tables are involved numerous parameters need to be fixed for the packages and this should be organized in a structured way. Therefore, we decided to collect the application parameters not simply as SAS macro variables like the Swedish SAS2Argus macro does. Our packages expect Excel workbooks with several work sheets providing information on files, folders, variable names, structures etc.

Once this parameter-file has been prepared and the application has been tested, starting it is straightforward and does not require the involvement of the SDC department during actual production. The design of a complex application is however far from easy. In that sense the packages are tools for SDC experts. Because preparing the Excel files can be somewhat cumbersome and error prone, we are working now on graphical user interfaces (GUI) making it easier to provide the necessary information to the system. Main output of the GUIs are the Excel parameter files[2] along with some application specific SAS macro code. Users can then add those macros to a SAS project. Executing such a project will activate our package (SDC tabulation or ARGUS control, resp.) with settings given by the parameter files.

In the following section we briefly recall the concept of the ARGUS control package itself. Sections 3 and 4 will present some screenshots of the GUI's and discuss the kind of information collected through those screens. The paper is completed by a short summary and some final remarks. Throughout the paper we assume some familiarity with the general concepts of tabular data disclosure control and with τ-ARGUS in particular.

[2] It should be noticed that not all parameters of the Excel files are accessible via the GUI. Some are considered as pre-defined. If necessary, they should be changed manually.

2 The traditional approach to coordinate secondary cell suppression in a set of linked tables

When tables are linked through simple linear constraints, cell suppressions must obviously be coordinated between tables. The most typical case is when tables share common cells (usually marginal totals), i.e., when they are linked through constraints saying literally that cell X of table A is identical to cell Y of table B. (Wolf and Giessing, 2008) investigated different approaches to deal with such problems. The best performing solution, referred to as "adapted modular", is meanwhile implemented in τ-ARGUS. Unfortunately, this method has certain limitations regarding the complexity of problems that can be handled. The technique implemented in our ARGUS control package has been referred to as "traditional approach" in that paper and offers more flexibility:

It is based on the idea of applying τ-ARGUS modular to each table within a set of linked tables separately within a backtracking procedure[3.] As in (Giessing, 2009) we refer to the backtracking procedure for a set of n tables $\{T_1, \ldots, T_n\}$ as 'simple linked tables sequence'. For the case of three linked tables T_1, T_2 and T_3 the approach is depicted in fig. 2.1.

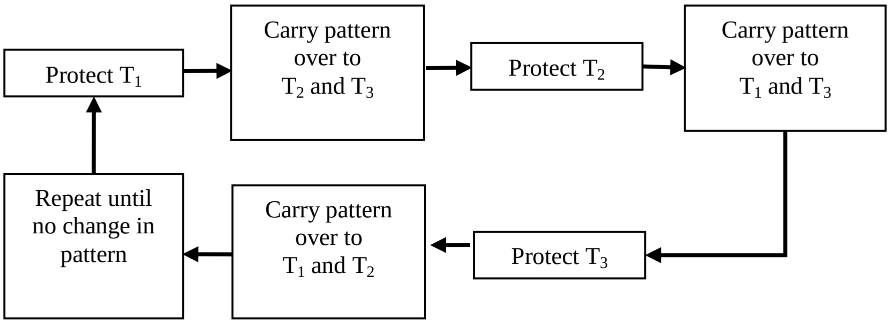

Protect T1
Carry pattern over to T2 and T3
Protect T2
Carry pattern over to T1 and T3
Protect T3
Carry pattern over to T1 and T2
Repeat until no change in pattern

Fig 2.1 Graphical representation of the 'traditional approach'

The sequence of protecting the tables has an effect on the amount and the pattern of secondary suppressions. If the number of tables is large, it is sometimes useful to form groups of tables for the linked tables processing[4]. This has also the advantage that we can apply different algorithms to different groups, e.g. using "adapted modular" where feasible, and "simple linked tables sequence" otherwise.

In order to achieve between-groups consistency we also need an "outer loop": After protecting a group of tables, new secondary suppressions are "carried over" to tables in groups processed earlier, and we start again from the beginning, e.g. with the first group of tables. Outer and inner loops are repeated until all tables of all groups are protected consistently and no new secondary suppressions get assigned during a full loop that would have to be carried over to any of the other tables processed earlier in the current loop. See (Schmidt and Giessing, 2010) for details.

[3] Backtracking procedures which carry over old secondary suppressions from one table to another are a typical way of implementing co-ordination of secondary suppressions between tables. The approach is used internally in the τ-ARGUS modular sub-table-backtracking. See also (Hundepool et al., 2012, sec. 4.4.4).

[4] See (Gießing 2009).

3 The GUI for the SDC Tabulation Package

Once a new or an already existing SDC tabulation project is opened, there are three different tabs, e.g. “Data”, “Classifications” and “Tables”. Sections 3.1 to 3.3 provide the respective screen shots along with some discussion.

One general idea of both GUI’s is that the user input will be stored as a kind of “project file” in the format of yet another Excel workbook, produced in addition to the Excel interface files required for the SDC tabulation and ARGUS control package. The GUI user can choose starting to work “from scratch”, or by “loading” an existing project which can then be modified or extended and saved again. E.g. after going through all tabs when pushing the “Save As” button a number of excel files will be produced. Besides a master excel file (the “project file”) storing all the information provided through the three tabs, a separate excel file will be produced for each table and stored in the folder specified in tab “Data” under “Folder Table Files” (c.f. fig 3.1) along with application specific SAS macro codes also generated by the GUI to be integrated in a respective SAS project as explained at the end of section 1.

3.1 Tab “Data”

Figure 3.1 presents the “Data” tab. After the user has specified the directory where the microdata file is stored[5], all SAS-files in this folder are available in the drop down box on the right side. Throughout both GUIs we expect users to assign libnames to all folders selected via the interfaces, in order to simplify the handling in SAS. Besides the path for the microdata two target folders (along with associated libnames) should be specified for the SDC tabulation package. One will be used by the package as place to store the SAS-format tabular data, the other one will be the place for the .hrc-files with the hierarchical structure of the explanatory variables.

At the bottom of the mask the response and the frequency variable[6] have to be chosen. In both cases all numerical variables of the selected microdata file are listed in a drop down box. Because in our experience some scaling can improve the efficiency of the optimization routines in the secondary suppression step, the SDC tabulation package offers to scale the microdata before tabulation[7]. Note that our scaling facility particularly “uprounds” microdata with a value below two (after scaling) to two[8].

Last thing to specify in the first tab is the parameter for the p%-rule[9]. Note that the tabulation package computes cell level data with and without scaling. The p%-rule is applied to the unscaled data[10].

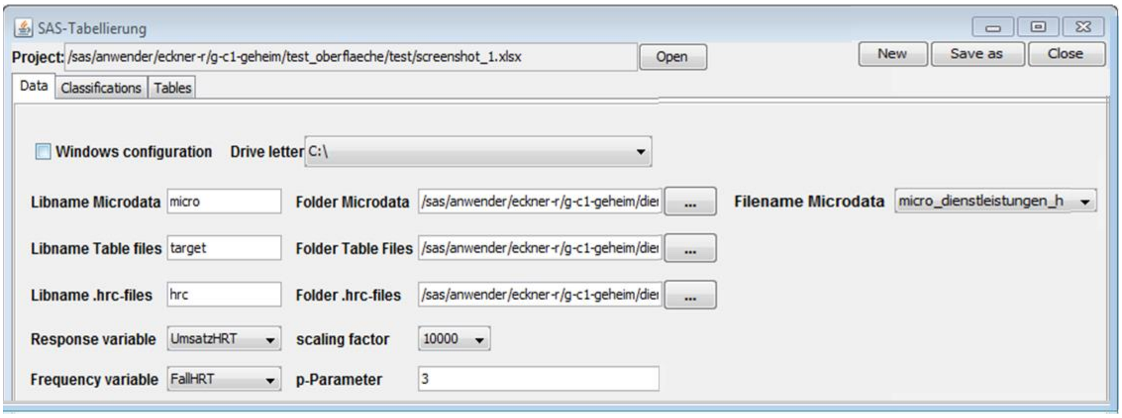
[5] All paths can be specified either in Windows or in Linux/Unix format.

[6] Response variable: the variable used to calculate the cell total; Frequency variable: Binary (0,1) variable used to calculate the number of observations making up the cell total.

[7] We recommend choosing the scaling factor at least large enough to guarantee that no cell value (including marginal cells and overall total) will be more than 10^{10} when computed on basis of the scaled data.

[8] Otherwise, there might be cells with a cell value of 1 or smaller which may cause certain problems during secondary suppression, when a table is protected for the 2nd (or more) time (in a backtracking step of a linked-tables process).

[9] According to a decision of the board of directors of the German Statistical Institues, the p%-rule should be preferred as concentration rule. Therefore, unlike ARGUS, our packages do not support (n,k)-dominance rules. See also (Hundepool et al., 2012, sec. 4.2.2).



Project: /sas/anwender/eckner-r/g-c1-geheim/test_oberflaeche/test/screenshot_1.xlsx

☐ Windows configuration Drive letter C:\

Libname Microdata: micro Folder Microdata: /sas/anwender/eckner-r/g-c1-geheim/die Filename Microdata: micro_dienstleistungen_h

Libname Table files: target Folder Table Files: /sas/anwender/eckner-r/g-c1-geheim/die

Libname .hrc-files: hrc Folder .hrc-files: /sas/anwender/eckner-r/g-c1-geheim/die

Response variable: UmsatzHRT scaling factor: 10000

Frequency variable: FallHRT p-Parameter: 3

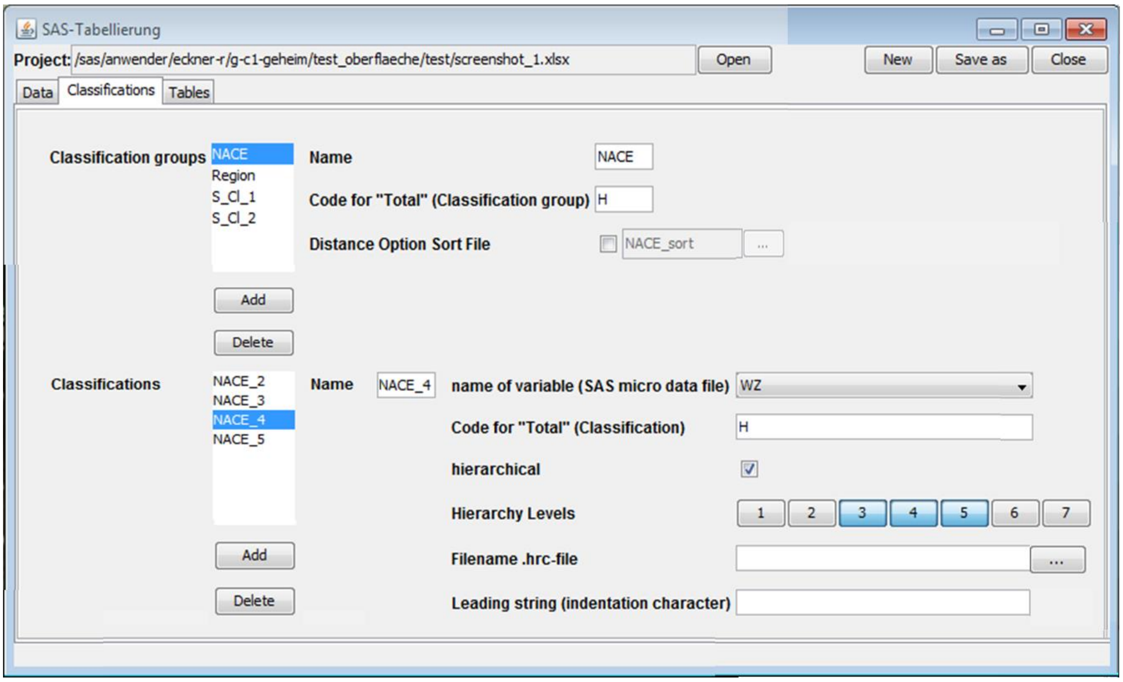
Fig 3.1 Screenshot of the “Data” tab mask.

3.2 Tab “Classifications”

Like ARGUS, our SDC tabulation package builds on the concept that a table is defined by the cross-combination of its spanning variables (together with the response variable). The GUI explicitly supports a generic approach to table definition (particularly: of linked tables) in the sense that it encourages the user to systematically “explore” the spanning variables, and to define multiple classifications for the same spanning variable in a structured way. E.g. we set up a structure, where all classifications relating to the same variable (f.i. Geography) form a “classification group”. In the upper pane on the left hand side of the tab (c.f. figure 3.2) the user can define such groups, assign a name to a group, and define the general domain of the respective spanning variable by specifying the code for the “Total” (NACE sector “H” in the instance presented in figure 3.2). As we often make use of the τ–ARGUS “distance function” (c.f. Wolf et al., 2014, 4.4.1) to influence the pattern of secondary suppressions in a special way, the GUI offers to specify certain settings for this special option. In the classifications tab, users can check the button “Distance Option Sort File” and choose an appropriate, existing file[11]. Doing so will generate macro code to merge the selected distance option sort file with the SAS format table files produced by the SDC tabulation application.

[10] Like ARGUS, our SDC tabulation package offers special options for the case of data from sample surveys, or for how to consider holding structures. Those options are foreseen to be supported by future versions of the GUI.

[11] Using the distance function makes ARGUS prefer a suppression pattern where suppressed cells are “close” to each other. While for “ordinal” classifications (typically: size class) this makes some sense directly, it is not obvious for other dimensions a table may have. We have developed an approach exploring the frequency of sensitive cells: for a hierarchical classification the idea is to sort child categories of the same parent node in such a way that categories with many sensitive cells (in a larger table) are “close” to each other. The method generates a new coding scheme for the variable to follow this order sequence. The “distance option sort file” is a SAS file linking the original codes to the new coding scheme.



Project: /sas/anwender/eckner-r/g-c1-geheim/test_oberflaeche/test/screenshot_1.xlsx

Classification groups: NACE, Region, S_Cl_1, S_Cl_2

Name: NACE

Code for "Total" (Classification group): H

Distance Option Sort File: □ NACE_sort

Classifications: NACE_2, NACE_3, NACE_4, NACE_5

Name: NACE_4

name of variable (SAS micro data file): WZ

Code for "Total" (Classification): H

hierarchical: ☑

Hierarchy Levels: 1 2 3 4 5 6 7

Filename .hrc-file:

Leading string (indentation character):

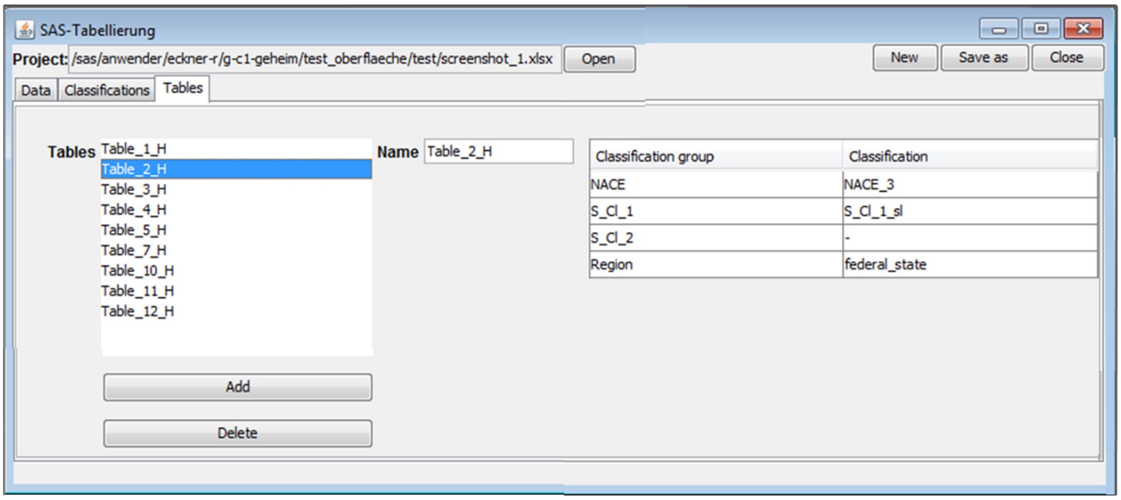
Fig 3.2 Screenshot of the “Classifications” tab mask.

Once a classification group is defined, specific classifications for this spanning variable can be constructed in the lower part of the mask. Fig 3.2 shows this for the variable “NACE”. In that example, four different classifications for NACE sector “H” are defined[12]. In the instance of fig. 3.2, NACE_4 is specified as hierarchical classification in a very similar style as in τ-ARGUS through the digits of the code for the microdata variable “WZ”. Like in ARGUS an alternative option is offered to derive a hierarchy by selecting an existing file (in ARGUS .hrc-format) with the description of the hierarchical structure. In that case the appropriate indentation character must to be declared.

3.3 Tab “Tables”

The third tab is for specifying tables. After pushing the “Add”-Button the user is asked to assign a name to a new table. In the left column of the pane on the right hand side of the mask, all the specified classification groups are listed. Classifications that obviously define the new table should now be chosen via drop down lists in the right column of this pane. Otherwise, the user selects a hyphen (‘-‘).

[12] Note that the total code of a specific classification may differ from the total of the classification group. In such a case, tables defined with this classification are tables for a sub-domain only.



Project: /sas/anwender/eckner-r/g-c1-geheim/test_oberflaeche/test/screenshot_1.xlsx

Tables:
Table_1_H
Table_2_H
Table_3_H
Table_4_H
Table_5_H
Table_7_H
Table_10_H
Table_11_H
Table_12_H

Name: Table_2_H

Classification group	Classification
NACE	NACE_3
S_Cl_1	S_Cl_1_sl
S_Cl_2	-
Region	federal_state

Fig 3.3 Screenshot of the “Tables” tab mask.

4 The GUI for the ARGUS Control Package

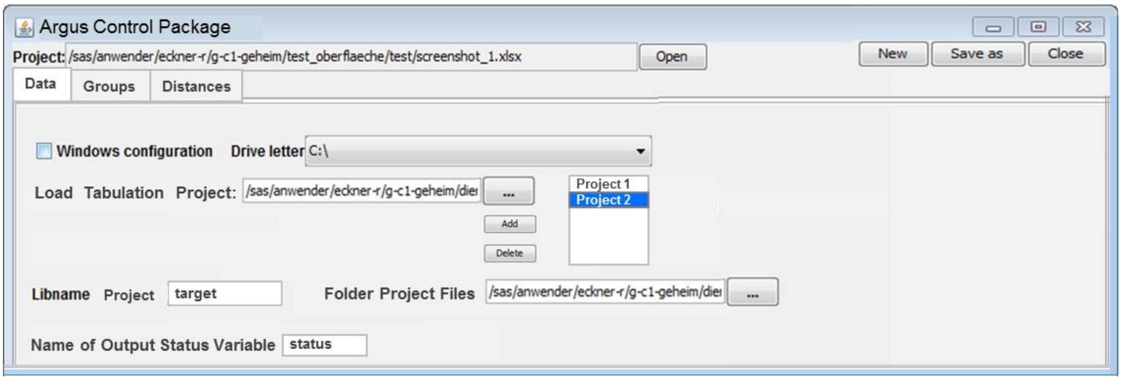
Following the approach of the first GUI, also the second GUI allows the users choosing to either work “from scratch” or to “load” an existing project to be modified or extended and saved again. In addition to this, it is also foreseen to read in one or more projects that have been developed with the SDC tabulation interface.

The following sections 4.1 to 4.3 explain the three tabs offered by the GUI for the ARGUS Control Package.

4.1 Tab “Data” of the ARGUS Control Package GUI

After choosing the appropriate setting for path formats (Windows/Unix) users are requested to “add” one or more SDC tabulation GUI projects. Usually, it will be only one, but sometimes we need some tricky processing that cannot be handled in a single SDC tabulation project. Like, when we need more than one (version of the) microdata-file to specify all tables of an application. However, in such a case consistency requirements must be checked. The user has to make sure that in all projects the specifications for the response and frequency variable, as well as the scaling factor are identical, and of course two tables with the same name are not allowed. These checks are validated automatically by the system.

At the bottom of the screen a libname and path have to be selected to specify the target folder for the Argus Control Package project files. Additionally, the user must specify a name for the variable that will indicate the final cell status (e.g. as unsuppressed, primary or secondary suppression) after processing.



Project: /sas/anwender/eckner-r/g-c1-geheim/test_oberflaeche/test/screenshot_1.xlsx

□ Windows configuration Drive letter C:\

Load Tabulation Project: /sas/anwender/eckner-r/g-c1-geheim/dier

Project 1
Project 2

Libname Project target

Folder Project Files /sas/anwender/eckner-r/g-c1-geheim/dier

Name of Output Status Variable status

Fig 4.1 Screenshot of the "Data" tab mask of the ARGUS Control Package GUI

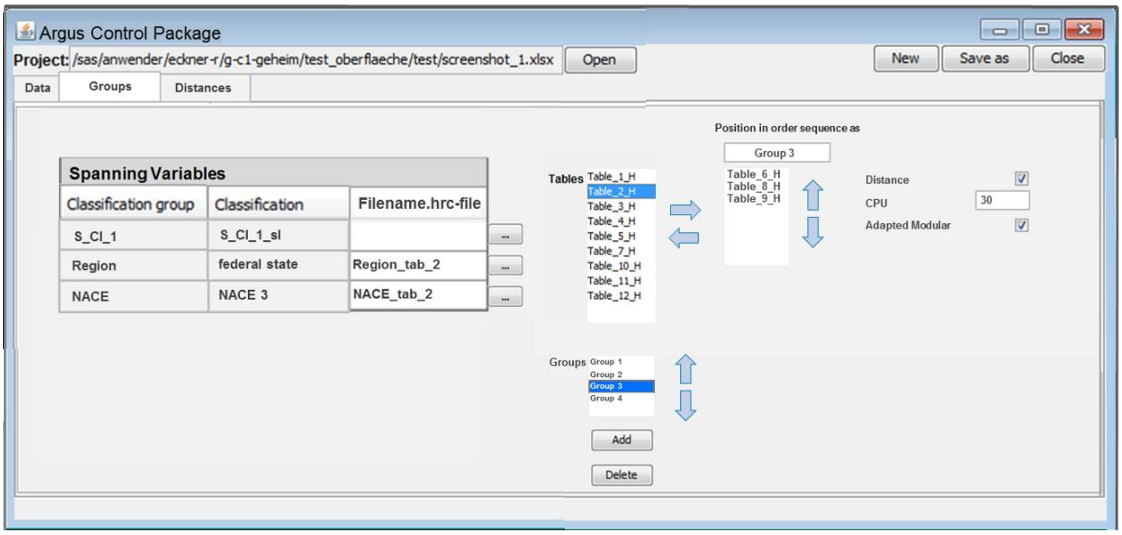
4.2 Tab "Groups"

The pane "Tables" in the centre of the tab lists all tables from all tabulation projects added to the current project. As mentioned in section 2, when using the traditional method to co-ordinate secondary suppressions across a large set of tables, it is sometimes useful to group them for the linked tables processing. This is supported by this tab.

After adding a new group to the list presented by the pane "Groups" (a group number will be assigned automatically) the user is supposed to move tables from the list of tables presented in the pane "Tables" to the pane on the right side with the list of tables for this group. For a selected table, the "spanning variables" pane on the left hand side provides background information on the structure of this table (e.g. the particular classifications and the name of the .hrc-file with the hierarchy structure of a classification[13]).

Using the arrow buttons, the user can always change the order sequence of the tables within a group as well as the order sequence of the groups. For each group three parameters relevant for the processing with the τ-ARGUS modular method have to be specified. The ARGUS control package will then prepare the ARGUS applications in such a way that the selected parameter is used for all tables within that group. Those parameters concern use of the distance function, the CPU time (in minutes) to be spent by the optimization routines on improving a feasible suppression pattern for a sub-table, and if the group should be protected using the ARGUS 'own' linked tables approach "adapted modular" (which is the recommended option, whenever feasible).

[13] Note that our SDC tabulation package automatically creates a .hrc-file for every classification variable, if not already supplied as input (c.f. sec. 3.2).



Project: /sas/anwender/eckner-r/g-c1-geheim/test_oberflaeche/test/screenshot_1.xlsx

Spanning Variables		
Classification group	Classification	Filename.hrc-file
S_Cl_1	S_Cl_1_sl	
Region	federal state	Region_tab_2
NACE	NACE 3	NACE_tab_2

Tables: Table_1_H, Table_2_H, Table_3_H, Table_4_H, Table_5_H, Table_7_H, Table_10_H, Table_11_H, Table_12_H

Position in order sequence as: Group 3

Table_6_H
Table_8_H
Table_9_H

Distance ☑

CPU 30

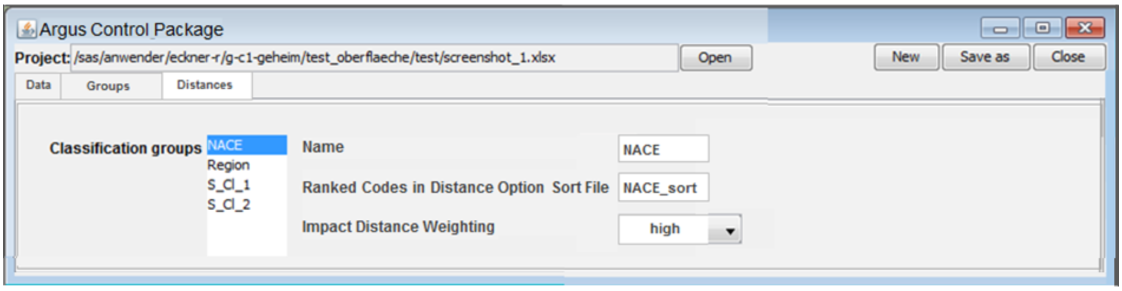
Adapted Modular ☑

Groups: Group 1, Group 2, Group 3, Group 4

Fig 4.2 Screenshot of the “Groups” tab mask

4.3 Tab “Distances”

The final tab is only relevant when using the τ–ARGUS distance function. In that case users specify for each classification group how strong the impact of this special cost function on the suppression pattern regarding this spanning variable should be. For example, we might be interested in a strong impact regarding a size class dimension, but not regarding NACE. Choices offered are “high”, “low” and “no”. This choice will be reflected in the ARGUS metadata prepared by the ARGUS control package[14], e.g. in all .rda-files relating to a classification of this classification group.



Project: /sas/anwender/eckner-r/g-c1-geheim/test_oberflaeche/test/screenshot_1.xlsx

Classification groups: NACE, Region, S_Cl_1, S_Cl_2

Name: NACE

Ranked Codes in Distance Option Sort File: NACE_sort

Impact Distance Weighting: high

Fig 4.3 Screenshot of the “Distances” tab mask

5 Summary and Final Remarks

In the context of integrating the SDC package τ-ARGUS into a production chain we have developed a SAS macro package especially suited for complex SDC processes involving multiple, linked tables. The package serves as control centre, executing

[14] Before using the distance function of τ-ARGUS, one must supply a cost scheme that defines cell costs depending on the number of steps a cell is away from another, already suppressed cell. For at most 5 steps of distance the costs can be specified by creating a parameter <DISTANCE> in the .rda-file that consists of a sequence of five numbers like 1 3 5 17 17, literally saying that a direct neighbour should have low costs (1) and cells four and more steps away from a primary get high costs (17). No impact of the distance would be implemented by a “constant” scheme like 17 17 17 17 17, whereas 7 17 17 17 17 for example leads to a slight preference of direct neighbour cells.

multiple ARGUS applications where the outcome of one application must be reflected in the input for the next. This ARGUS control package is supported by a second SAS package for the SDC tabulation process. While with these tools the SDC step can be easily integrated into a production chain, the design of complex applications is still often far from simple and a task for SDC experts. When many tables are involved, numerous parameters need to be fixed.

Therefore we are developing now graphical user interfaces for both tools. While this is still work in progress, the paper has described the different tab masks and the information to be captured by them. An advantage of having those GUIs might be that it may eventually make it easier in the future to share our tools with other SDC experts who are familiar with both, SAS and τ-ARGUS.

Regarding complex applications, a particular strength of the GUI for the tabulation package is that it enforces a systematic definition of multiple (where needed) classifications for the spanning variables, because this makes the structure of links between tables more obvious. Perhaps the suggested approach can be taken as an idea for the future development of τ-ARGUS which should anyway be extended in the long run by a more general algorithm for coordination of cell suppression between linked tables.

References

Almberg, L.-E., Andersson, K., Sun, L. (2013), '*Experiences of implementing Bifrost*', paper presented at the Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality (Ottawa, 28-30 October 2013) available at http://www.unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.46/2013/Topic_5_Sweden.pdf

Giessing, S. (2009), '*Techniques for Using τ -Argus Modular on Sets of Linked Tables*', paper presented at the Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality (Bilbao, 2-4 December 2009) available at http://www.unece.org/stats/documents/2009.12.confidentiality.htm

Giessing, S., (2013), '*Software tools for assessing disclosure risk and producing lower risk tabular data*', Report, Deliverable D11.1 – Part B of Project N°: 262608 "Data without Boundaries", available at http://www.dwbproject.org/export/sites/default/about/public_deliveraples/dwb_d11-1b_software-tools-disclosure-risk-assessment.pdf

Hundepool, A., Domingo-Ferrer, J., Franconi, L., Giessing, S., Schulte Nordholt,E., Spicer, K., and Wolf, P.P. de (2012), *Statistical Disclosure Control*, Wiley, Chichester, United Kingdom.

Kraftling, A. (2011). *SAS2Argus user manual*. Unpublished manuscript. Statistics Sweden.

Schmidt, K., Giessing, S. (2010), '*Techniques for Using τ -Argus Modular on Sets of Linked Table-SAS implementations*', paper presented at the Privacy on Statistical Databases conference (Corfu, 22-24 September 2010) available at Companion CD Proceedings, ISBN: 978-84-693-4265-7.

Schmidt, K., Giessing, S (2011). *A SAS-Tool for Managing Secondary Cell Suppression on Sets of Linked Tables by τ-ARGUS Modular*, Paper and poster presented at the NTTS 2011 in Brussels, February 2011, available at http://neon.vb.cbs.nl/casc/ESSNet2/Appendix3APaper_Schmidt.pdf

Wolf, P.P. de (2002), ‘*HiTaS: A Heuristic Approach to Cell Suppression in Hierarchical Tables*’, In: ‘*Inference Control in Statistical Databases*’ Domingo-Ferrer (Ed.), Springer (Lecture notes in computer science; Vol. 2316)

Wolf, P.P. de and Giessing, S. (2008), ‘*How to make the* τ *- Argus Modular Method Applicable to Linked Tables*’, in *Privacy in Statistical Databases,* Domingo-Ferrer and Saygin (Eds.), Springer LNCS 5262, pp 37-50

Wolf. P.P. de, Hundepool, A.J., Giessing, S., Salazar, J.J., and Castro, J. (2014), *τ-ARGUS User's manual*, Statistics Netherlands, The Hague.