UNECE

CONFERENCE OF EUROPEAN STATISTICIANS

**Expert Meeting on Statistical Data Confidentiality**

1-3 December 2021, Poland

# Introducing a graphical user interface for creating the metadata governing the secondary cell suppression process.

Michel Reiffert (Destatis)
*Michel.Reiffert@destatis.de*

***Abstract***

Many statistics in the German Statistical System produce large sets of multiple linked tables. In order to protect those sets of tables correctly it is usually necessary to run the τ-ARGUS Modular method multiple times. Furthermore, the results of previous runs can affect the input of the following calculations. Fortunately, there exists a set of SAS macros (referred to as "Argus Control package" in the following), which handle this task in an automated and efficient manner (Schmidt and Giessing, 2011).

All parameters and metadata governing this process are saved as excel files, and as of now have to be edited manually by the user. The process of creating or editing the metadata files is error prone and in some respects not intuitive. Therefore, the idea of developing a user-friendly GUI for this tool has already been discussed in (Giessing and Grunwald, 2015).

The present paper provides a detailed insight into the GUIs for creating the metadata excel files in an easy to use and somewhat more intuitive way. In particular, the paper extensively describes the implementation and the possible uses of the GUI for the metadata/parameter file as interface of the Argus Control package steering the secondary suppression process with τ-Argus applied to each table or set of linked tables of a potentially large collection of multiple linked tables.

# Introducing a graphical user interface for creating the metadata governing the secondary cell suppression process

Michel Reiffert*, Sarah Giessing**, Sven Grunwald***

* Statistisches Bundesamt, 65180 Wiesbaden, Germany, Michel.Reiffert@destatis.de

** Statistisches Bundesamt, 65180 Wiesbaden, Germany, Sarah.Giessing@destatis.de

*** Statistisches Bundesamt, 65180 Wiesbaden, Germany, Sven.Grunwald@destatis.de

**Abstract**: Many statistics in the German Statistical System produce large sets of multiple linked tables. In order to protect those sets of tables correctly it is usually necessary to run the $\tau$-ARGUS Modular method multiple times. Furthermore, the results of previous runs can affect the input of the following calculations. Fortunately, there exists a set of SAS macros (referred to as "Argus Control package" in the following), which handle this task in an automated and efficient manner (Schmidt and Giessing, 2011).

All parameters and metadata governing this process are saved as excel files, and as of now have to be edited manually by the user. The process of creating or editing the metadata files is error prone and in some respects not intuitive. Therefore, the idea of developing a user-friendly GUI for this tool has already been discussed in (Giessing and Grunwald, 2015).

The present paper provides a detailed insight into the GUIs for creating the metadata excel files in an easy to use and somewhat more intuitive way. In particular, the paper extensively describes the implementation and the possible uses of the GUI for the metadata/parameter file as interface of the Argus Control package steering the secondary suppression process with $\tau$-Argus applied to each table or set of linked tables of a potentially large collection of multiple linked tables.

## 1. Introduction

Some agencies have developed wrapper packages to integrate $\tau$-ARGUS into their production process, for example one approach described in (Almberg et al., 2013). As described in (Giessing and Grunwald, 2015) and in detail in (Schmidt and Giessing, 2011), we use a set of SAS routines to control the cell suppression for multiple, linked tables. This process consists of SDC tabulation as well as the use of $\tau$-ARGUS for secondary cell suppression, hereafter referred to as SDC tabulation and ARGUS control package, respectively.

The SAS macros of the SDC tabulation and the ARGUS control package both start by reading a metadata file which specifies all the necessary parameters. In the tabulation process we need for each (sub)-table a metadata file, whereas the ARGUS control package needs only one metadata file for all (sub)-tables that can be grouped together for one (or multiple) $\tau$-ARGUS runs. All parameters and metadata governing this process are saved as excel files, and as of now have to be edited manually by the user. The process of creating or editing the metadata files is error prone and in some respects not intuitive. Therefore, the idea of developing a user-friendly GUI for this tool has already been discussed in (Giessing and Grunwald, 2015). The present paper introduces the current status of the implementation of the User Interfaces, focusing on the GUI for the ARGUS control package.

The following section provides the reader with methodological background information that clarifies the motivation to implement the GUIs as a user-friendly way to replace manual, and therefore error prone processes.

In Section 3, we give a brief overview of the Workflow of the GUIs and Interaction with the SAS/ τ-ARGUS environment. Furthermore, we show, by means of a diagram, how the GUIs are implemented into the SDC tabulation and the ARGUS control package processes.

In section 4, we take an in-depth look at the GUI for the ARGUS control package. In addition to some screenshots, we also explain the functions of the GUI and the resulting possibilities for the user. This is done based on an exemplary workflow, broken down into separate steps explaining all elements on the GUI and the functionalities. For this purpose, we refer to the example from (Schmidt and Giessing, 2011) and show how this specific set of tables is handled using the GUIs.

Section 5 gives a brief outlook on possible future developments to make the GUIs easier to use and to add more functionalities to them. The paper concludes with a summary and some final remarks.

## 2. Background and motivation

It is common for statistical agencies to release large sets of multiple linked tables. The cell suppressions must then be coordinated between all linked tables based on the same dataset. The software package $\tau$-ARGUS has very efficient algorithms implemented for dealing with linked tables, as described in (De Wolf et al., 2014).

However, it is not possible to run every set of tables as linked tables in $\tau$-ARGUS directly. There are two main reasons why this might be the case, namely dimensionality and incompatible hierarchy structures.

For example, consider the case of two hierarchical 3-dimensional tables that have one matching and two differing spanning variables. The cover table build by $\tau$-ARGUS when these two tables are put into a linked run is 5-dimensional. This might exceed the limit regarding maximum number of dimensions for working with the Modular method. In practice, we encountered problems with the above-described example using $\tau$-ARGUS Version 3.5 build 27.

Furthermore, it is assumed that tables that use the same spanning variables only have hierarchies that can be covered by a single hierarchy for that spanning variable. This is not necessarily the case for all spanning variables. For more detailed information on hierarchies, we refer the reader to (De Wolf, 2007).

Therefore, it is necessary to implement a process to coordinate the suppression patterns while using $\tau$-ARGUS. We use a working solution that handles sets of linked tables by applying the $\tau$-ARGUS modular method to each single table or groups of tables as a module within an iterative procedure, which is introduced in (Schmidt and Giessing, 2011) and referred to as “traditional approach” in (De Wolf and Giessing, 2008). As mentioned in the introduction, this process is controlled by a set of metadata files. The motivation for creating the GUIs stems mostly from the fact that these files have to be created and filled manually. To get an idea of the time-consuming and error prone work that is involved in that, see Appendixes A and B for screenshots of the Excel files for the SDC tabulation and the ARGUS control package respectively.

The remainder of the paper will focus on the implementation and functionalities of the ARGUS control package GUI.

## 3. Interaction of the GUIs and the SAS/ $\tau$-ARGUS environment

In this section, we explain which objects and data types play a role in the SDC process and how they are linked to each other via the GUIs. This is to give the reader a better overview of the SDC process, and a better understanding of the focus of this paper.

The following diagram provides a schematic overview of the SDC processes, as well as the workflow of the GUIs and Interaction with the SAS/ τ-ARGUS environment:

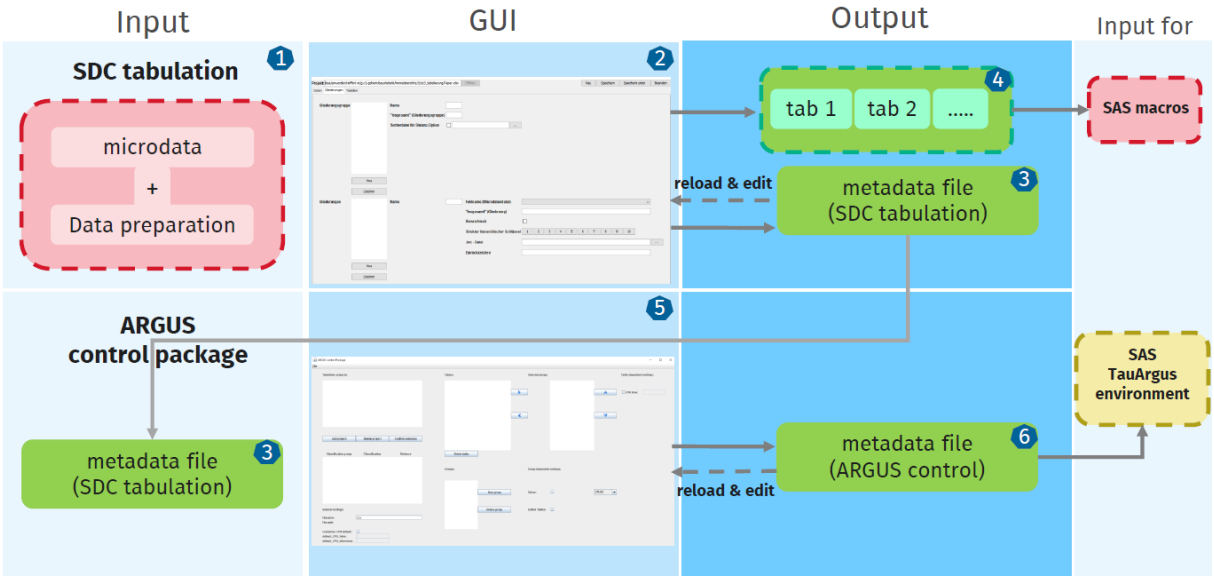



**Fig 1.** Workflow of the GUIs and Interaction with the SAS/ $\tau$-ARGUS environment

In this diagram, a number has been assigned to each object (f.i. the ARGUS control package GUI is denoted by (5)). The colour system indicates from which IT environment the associated object originates. Green represents Excel files; pink represents files and information coming from the SAS environment.

The upper level (horizontally) shows the SDC tabulation process with the GUI (2) in the center. The user provides input to fill the SDC tabulation GUI as well as the microdata, in our case in SAS format (1), which specifies the dimensions and some more details about the dataset. For a more in depth analysis and description of the SDC tabulation GUI, we refer the reader to (Giessing and Grunwald, 2015).

The GUI (2) then produces two different types of output. Firstly, it saves all the relevant information in the metadata file (3), which is, at the same time, output of the SDC tabulation GUI and input for the ARGUS control package GUI. Secondly, it writes an Excel file for each table, represented collectively in the diagram by (4). Those files are used directly by the SAS macros we use to control the SDC tabulation process.

The lower half of the diagram analogously shows the ARGUS control package GUI (5) and the interface with the SAS/ τ-ARGUS environment constituted by the metadata file (6). The remainder of this paper focuses on how to create the metadata file (6) via the GUI (5).

As can already be seen at this point, essentially the GUIs automate the process of creating the files (4) and (6). We will later demonstrate in this article, that this approach saves a lot of manual work.

## 4. GUI for the ARGUS control package

Generally, there are two ways of working on a project with the ARGUS control package GUI. One way is to start a completely new project, the second way is to load an existing project and edit it. In this chapter, we will focus on the first way, because going through this process helps to illustrate, how the GUI is used.

Creating a project from scratch involves a couple of steps. While they do not have to be carried out in the exact order given below, those steps touch all aspects of working with the GUI and explain all functionalities. Before we look at every single step in-depth and explain all settings and functionalities, we give a brief overview of the process:

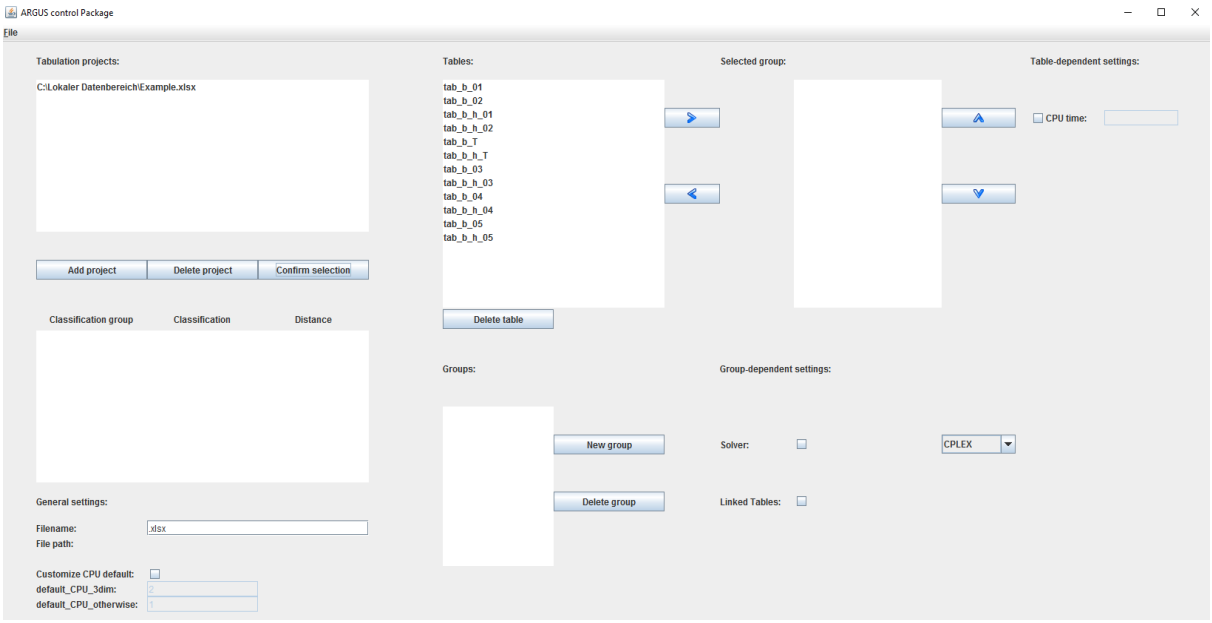
- Step 1: Loading tabulation project(s)
- Step 2: Set the general settings
- Step 3: Create the desired group structure
- Step 4: Set the group dependant settings for each group
- Step 5: Set the table dependant settings for each table
- Step 6: Save the project and create the metadata file
- Step 7: Load and edit the project

As mentioned in the introduction, we illustrate each of the above steps using a reduced version of the example from (Schmidt and Giessing, 2011). For the sake of clarity, we focus on a set of tables from five subregions. The following gives a brief overview over the table programme:

The German Länder (subregions) release a one-dimensional table “tab_b” and a two-dimensional table “tab_b_h” and accordingly “tab_b_T” and “tab_b_h_T” on the federal level. Where b is the explanatory variable, which is the NACE-code in a breakdown to the divisions. The table tab_b_h is defined by the explanatory variables b (NACE-code in breakdown only to the level of sections), and h (a size-class variable).

Step 1: Loading tabulation project(s)

The tabulation project window is located in the upper left corner of the GUI. By clicking the button “Add project” the user is able to select a metadata file from the SDC tabulation process. Note, that in certain cases it might be necessary to select more than one tabulation project, which is, of course, also possible. Once all the projects are loaded into the tabulation projects window, by clicking the “confirm selection” button all tables included in those tabulation projects will appear in the Tables window to the right of the tabulation projects window.

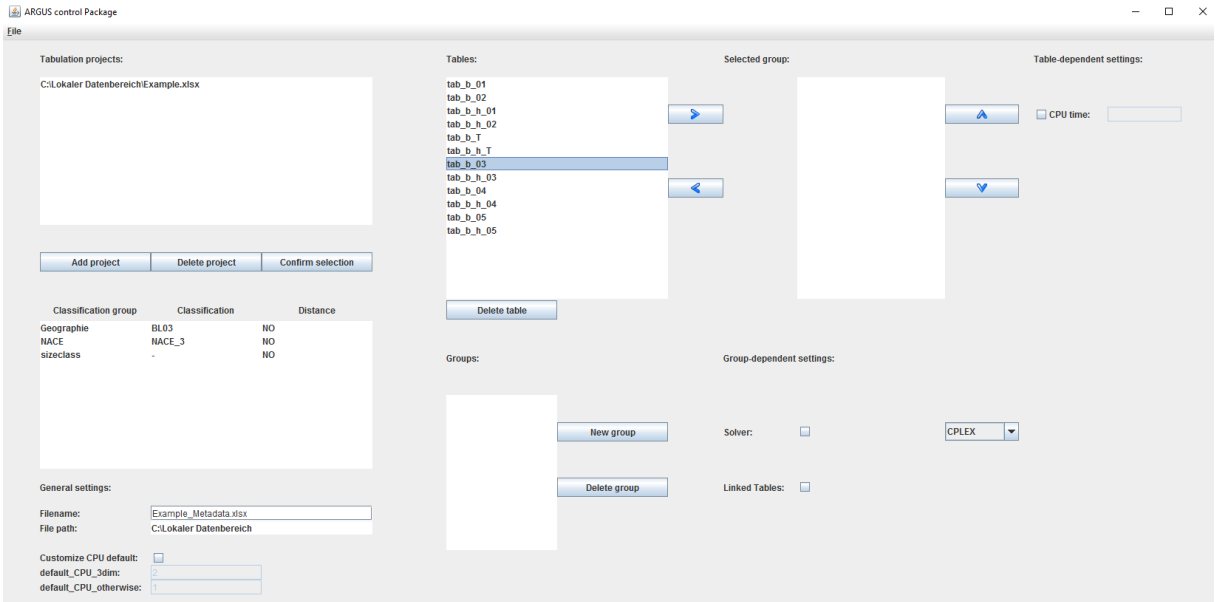



## Step 2: General settings

The general settings are located in the lower left corner of the GUI, and consist of the following parameters:

- Filename
- File path
- Customize CPU default

The filename for the output file of the GUI (i.e. the metadata file) is specified manually, and has to end in the pre-set file extension. File path is filled automatically once the project is saved via the "File>>Save file as" menu tab, and will show the path to the folder on the local computer or network.



Furthermore, the user can change the default setting for the CPU time used by $\tau$-ARGUS for three dimensional tables or tables with a different number of dimensions. As long as the checkbox remains unchecked the $\tau$-ARGUS default values of 2 minutes per three-dimensional table and 1 minute per other table stay in place and are visible in the greyed out text fields. By activating the checkbox, the user can input any positive integer in the now highlighted text fields.

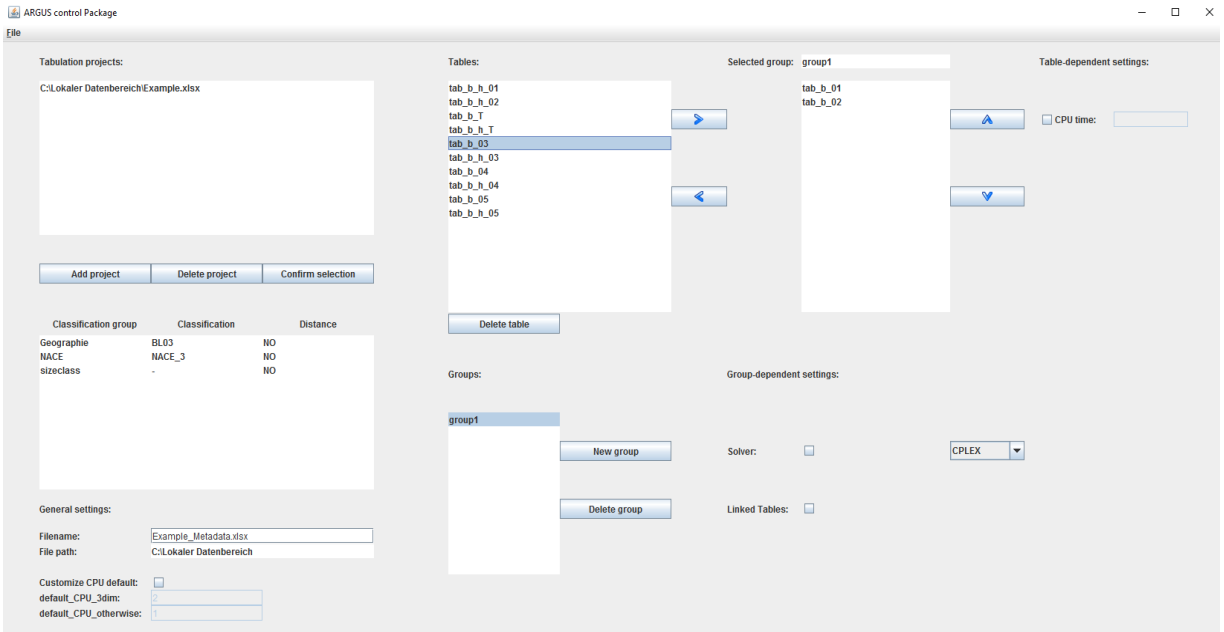
Step 3: Create the desired group structure

This step is the central building block of the application and therefore extends over multiple interacting windows of the GUI. Precisely, these are the three most central panes, namely "Tables", "Groups" and "Selected group".

First, the user can create a new group in the "Groups" window by clicking the "new group" button located on its right. The groups are automatically named "group1", "group2", etc. in ascending order, and cannot be manually renamed. This is for the sake of reducing potential errors, because the SAS macros expect the group names in this specific format.

The newly created group can now be selected by clicking. If carried out like this, the "Selected group" text field changes promptly from the previously selected group (or a blank, if it is the first group that was created) to the newly selected group. Of course, groups can be deleted by the "Delete group" button at any time and the existing groups will be renamed according to the ascending order.

Second, in the "Tables" window, the user can see all tables, which are available, because they have been loaded into the GUI as described in Step 1. To add one of these tables to the "Selected group" window, the user has to click the right arrow symbol between the "Tables" and "Selected group" windows. Intuitively, this process can be reversed by clicking the left arrow symbol.




It is also worth mentioning that the order of the tables inside of one group can be changed using the up and down arrows located on the right hand side of the "Selected group" window. This changes the order in which $\tau$-ARGUS protects the tables.

The user can now build the desired group structure in an easy to understand and visualized way, by merely repeating the above-described process. The “table overview window” consisting of “Classification group”, “Classification” and “Distance” is located on the left hand side of the GUI under the “Tabulation projects window”. It gives a structured view of the dimensions of the selected table, and therefore helps the user to keep an overview of the variables involved in the selected tables.

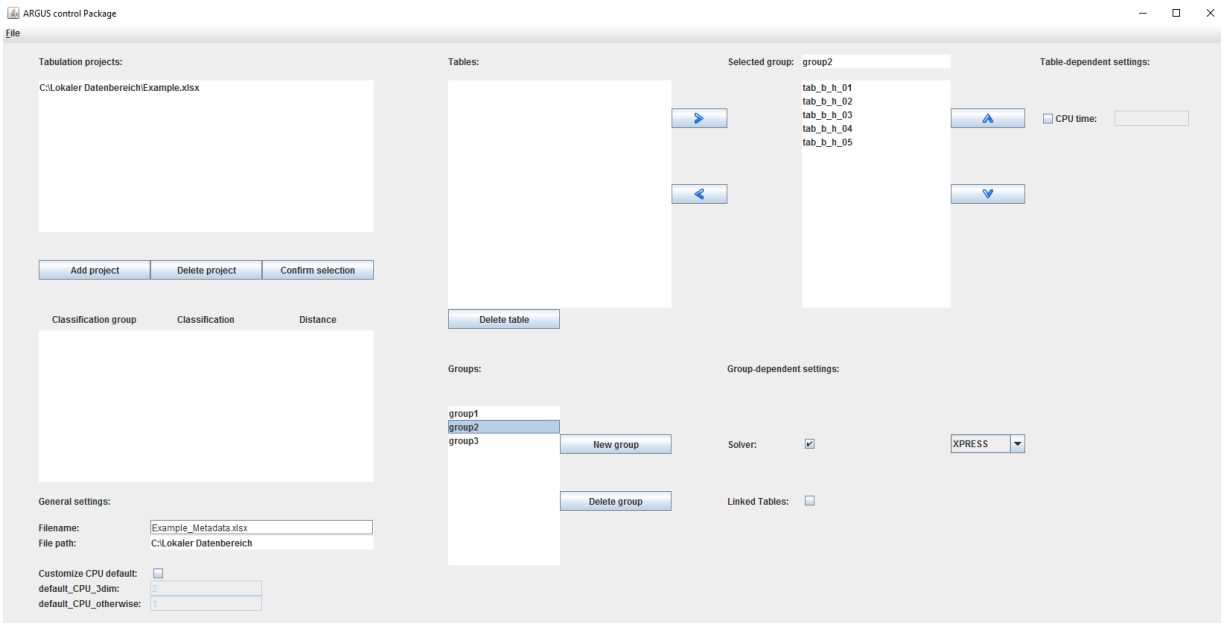
Step 4: Group dependant settings for each group

With the desired group structure in place, the user can set the group dependant settings for each group.

Firstly, the solver for the optimization problem that will be used by $\tau$-ARGUS can be specified. As of now, the GUI offers the choice between CPLEX, XPRESS and a Free solver (SCIP using SoPlex for solving the linear optimization programs). The choice for this setting will obviously depend on the user’s availability of solvers as well as preferences.

Secondly, the user can specify if the tables in the group are dealt with as a set of linked tables by $\tau$-ARGUS (as described in (Wolf et al., 2014)), or by the “simple linked tables” method of (Wolf and Giessing, 2008).

Both options can be activated by checking the respective checkbox and, in the case of the solver, select accordingly from the drop-down menu.




Step 5: Table dependant settings for each table

For each table the user can specify the CPU times in minutes that are allocated to solve this particular optimization problem, given by the selected table. Note that this setting is different from the “Customize CPU default” setting, as it specifies the CPU time for just one table. All tables, that are not specified, will be solved within the default CPU time.

To specify the CPU time, activate one table in the “Selected group” window by clicking, and then check the “CPU time” checkbox and type an integer into the now highlighted text field.

Step 6: Save the project and create the metadata file

Once all the steps have been carried out and all settings are fixed, the metadata file can be written as the output of the GUI. Note, that if it is the first time of saving the project, the user needs to save via the "File>>Save file as" menu tab. After that the user can save to the specified folder by simply using the "File>>Save file" menu.

Step 7: Load, edit or start a new project

Of course, it is also possible to load an existing project and edit the information, which is done by navigating to "File>>Open file". To start a new project from scratch, use the "File>>New project" menu tab, which will delete all existing entries. When using this option, any progress, that is not yet saved to a metadata file yet, will be lost.

## 5. Further Developments

Sections 3 and 4 presented the current functionalities of the two interfaces and explained how to use them by means of an example. This section provides an outlook on two possible further developments.

While the two interfaces provide the user with a clear and easy-to-use tool, there is no feedback during the process of filling the interfaces. This means that possible errors can only be discovered after a SAS/ $\tau$-ARGUS run. In order to be able to give interactive feedback, we would like to insert an active plausibility check during runtime for the ARGUS control package GUI. For example, an impermissible combination of dimensions in the group composition could be automatically detected and reported back to the user. Furthermore, the programme could issue warnings concerning, the dimensionality of the tables and related problems with ARGUS runtimes.

Additionally, in order to make the GUI useful to a broader user group, we plan to integrate a functionality that automatically prepares all settings, and hence all necessary Excel sheets for the next reporting period. The idea behind this is that the initial set up is developed by an SDC expert and tested for the first reporting period. Afterwards, such a functionality makes it possible to hand over the project to the statistics department. As long as the set of tables and the structure of the disseminated tables remain unchanged, the settings of the previous reporting period can be transferred to the next, while only certain paths and names have to be changed.

## 6. Summary and Final Remarks

This paper has presented a detailed overview of the SDC processes, as well as the workflow of the GUIs and interaction with the SAS/ τ-ARGUS environment. Additionally, it has explained, in a step-by-step guidance, how to use the GUI for the ARGUS control package by presenting a series of screenshots.

Following the method proposed in (Schmidt and Giessing, 2011), the user has to create and fill all Excel metadata sheets for the SDC process manually. Note, that for the described set of tables this involves 12 sheets for the SDC tabulation process and 1 more sheet for the ARGUS control package. Using the GUIs those steps are no longer necessary and all the sheets are automatically created as output.

Additionally, the risk of mistakes by manually creating the sheets is reduced considerably. Although, not all possible settings and parameters are controllable through the GUI, it is still

possible for the SDC expert to change those values manually in the already created sheets if necessary.

We believe that this software will be able to facilitate the work of SDC experts, and at the same time make the design of a complex SDC process for multiple linked tables more accessible to a wider range of users.

## 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

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

De Wolf, P.P. (2007), *'Cell suppression in a special class of linked tables'*, paper presented at the Joint ECE/Eurostat Worksession on Statistical Confidentiality in Manchester, December 2007, available at https://ec.europa.eu/eurostat/portal/page?_pageid=3154,70730193,3154_70730647&_dad=portal&_schema=PORTAL

De Wolf, P.P. 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

Giessing, S., Grunwald, S. (2015), *'A Graphical User Interface to Manage Cell Suppression on Sets of Linked Tables Using SAS and τ-Argus'*, paper presented at the Joint UNECE/Eurostat Work Session on Statistical Data Confidentiality (Helsinki, Finland, 5 to 7 October 2015) available at https://unece.org/fileadmin/DAM/stats/documents/ece/ces/ge.46/20150/Paper_20_Session_5_Germany__Giessing___Grunwald_.pdf

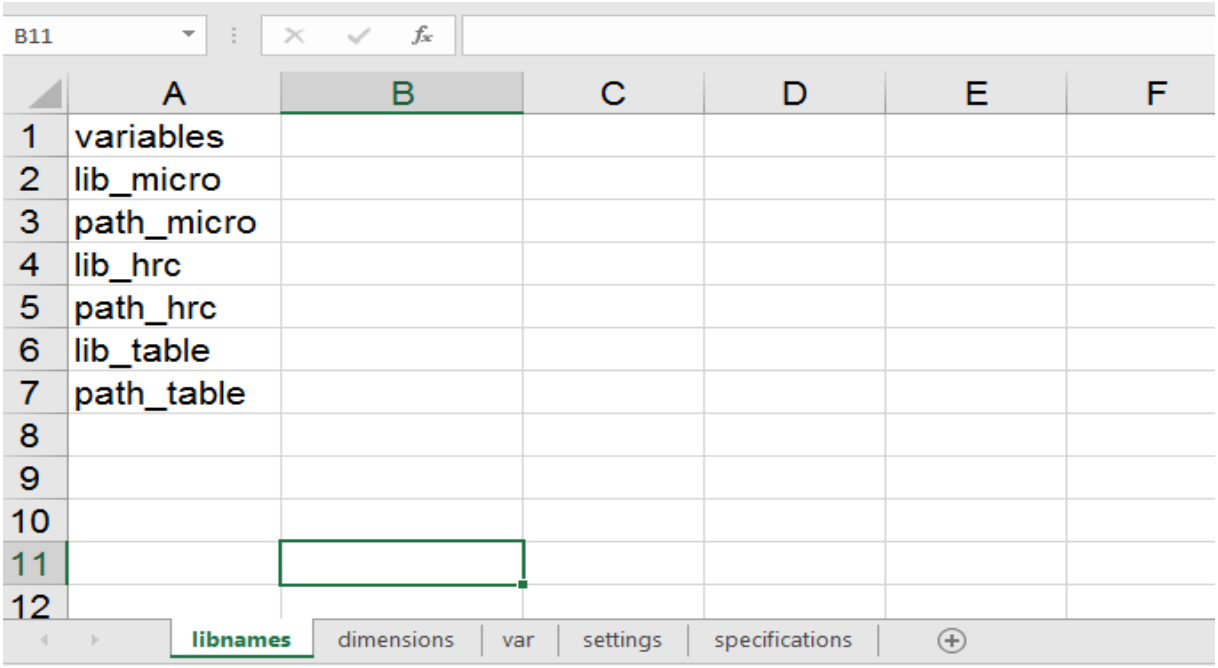
Schmidt, K., Giessing, S. (2010), *'Techniques for Using τ -Argus Modular on Sets of Linked Table-SAS implementations'*, paper presented at the Privacy on Statistical Databases conference (Corfu, 22-24 September 2010) available at Companion CD Proceedings, ISBN: 978-84-693-4265-7.

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

## Appendix A - SDC tabulation metadata file

To give an idea of the amount of manual work associated with creating the metadata files we present screenshots of the excel files that need to be created in order to handle the set of tables from the example given in this paper.

For each of the 12 tables the following excel file has to be filled with information and specifications of the table:




## Appendix B - ARGUS control package metadata file

The following excel file has to be filled only once for the ARGUS control package process, but for each of the 12 tables there exists a sheet in which the dimensions have to specified.