DESTATIS
Statistisches Bundesamt

Tau-Argus: progress and new ideas

SDC-tools User Group Meeting 2025

Sarah Giessing (Destatis)

Part I:
Tau-Argus – A case study in sharing services

Extracts (with modifications) from a presentation of

Lucas Quensel-von Kalben (Destatis) at the IT Working Group in April 2024

Functional design of Tau-Argus

2.16 *Functional design of τ-ARGUS*

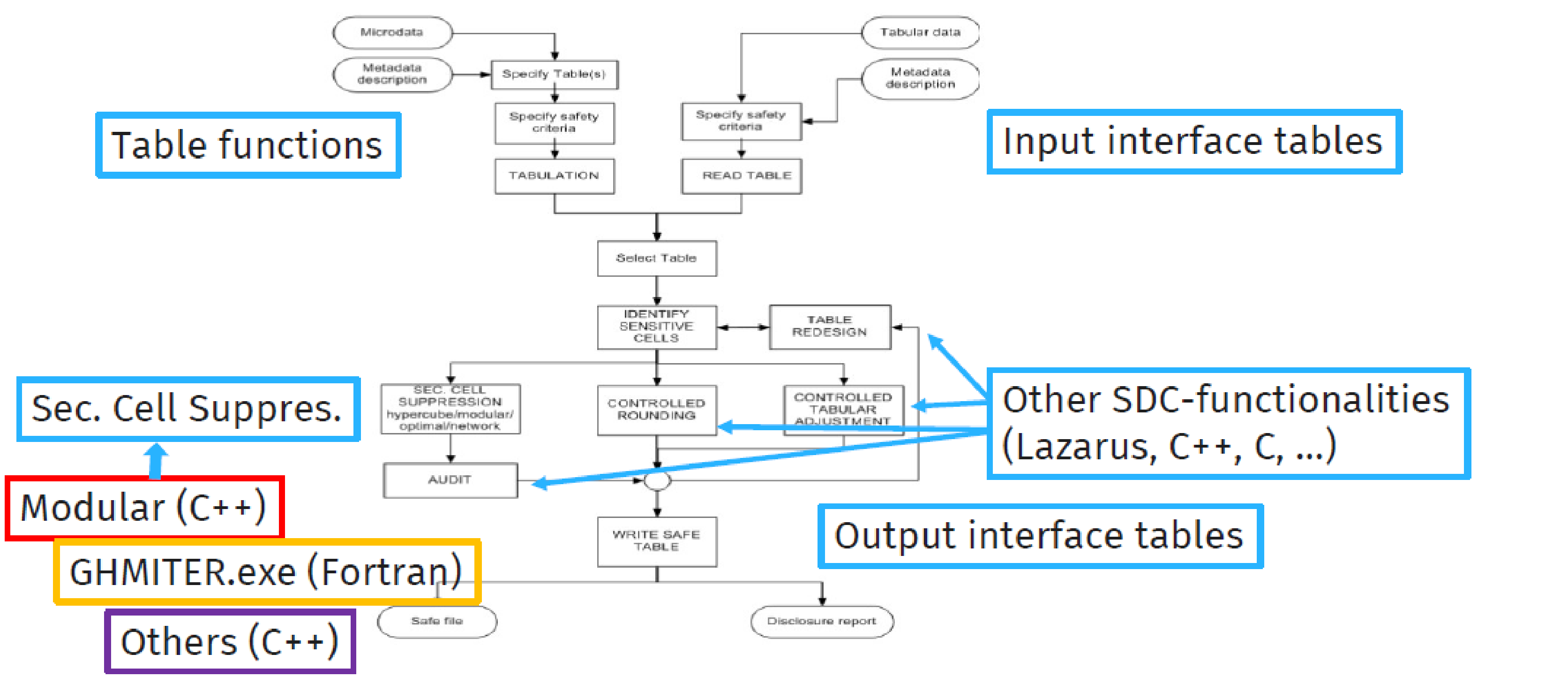

Table functions

Input interface tables

Sec. Cell Suppres.

Modular (C++)

GHMITER.exe (Fortran)

Others (C++)

Other SDC-functionalities (Lazarus, C++, C, ...)

Output interface tables

Tau-Argus key features

Advantages

- Modular: Many SDC-methods are already integrated
- Automatable: Capable of being integrated in common workflows (SAS, R, Java) via batch-language
- Confidentiality is preserved – Tau-Argus is ESS approved standard tool
- Cheap in production: (users only pay cost of solver-software, can even try to do without)
- Open Source: flexible to adapt; security by transparency

New functional requirements

- Potentially: Future request for integration of new or altered methodologies
- Process Integration: In production Tau-Argus is often not used at its own:
  - Currently: Integration by using batch language
  - In future: Develop API/contract to allow SDC methods to be called from “any” language/interface (e.g., Java, Python, R, cloud-based, web-based, ...)

New functional requirements

- In future: clear separation of GUI and calculation functionalities → allows direct call of calculations
  - Then: for calling a pre-designed Tau-Argus application from a production process, the GUI functionality is not needed
- Very important:
  - Fully ensure that calling a functionality/calculation via the GUI, or directly always gives exact same results!
- Otherwise:
  - Pre-designed Tau-Argus applications (via GUI) may give different results when called directly
  - Tau-Argus GUI not useable then for checking such (and other) unexpected results

Some Technical Issues

(Software Architect perspective)

- When installing Tau-Argus locally, the (Java, C++, Fortran) components / runtimes will be installed locally too. Such local installations may become unavailable in the next generation user desktop environment.
- There are open known vulnerabilities affecting the Java version and distributed runtimes.
- Code is shipped without signatures and cannot be verified to be trustworthy. Building the code is hard or impossible, because the build system is hard coded to work on a single machine (or not present at all in the case of GHMIter)
- Code is hard to understand
- There are no automated tests

Nevertheless: Tau-Argus is the main tool for Statistical Disclosure Control in Europe!

Lessons learned (not just Tau-Argus!)

- Shared services have a lot of challenges, f.i.
  - Updating and maintaining the code base
  - Securing production
  - Supporting the product (and provide trainings)
- Open Source is not an „easy sledding“ but has to be organised and sponsored

Idea presented at the IT Working Group (*though not taken up by the WG*):

- „organized“ = „organized in products“ (…rather than in projects)
- „sponsored“ = cost sharing between users (either by money or by providing developer capacities)

Contact

Statistisches Bundesamt
Statistische Geheimhaltung
65180 Wiesbaden

Sarah Giessing
Sarah.Giessing@destatis.de
phone: +49 611 75-2701

www.destatis.de

www.destatis.de/kontakt