Production of social statistics…goes social!

**Keywords:** production chain, statistical system.

# Objectives

The scope of this paper is to present a framework adopted in practice for the integration of software applications into a statistical production chain. The focus is the actual implementation of a high-level collaborative platform aiming not only at producing social statistics, but also at further fostering experimentation and analysis in that field. In doing so, we strongly support the (obvious) claim of **[1]** that *"the modernisation and industrialisation of official statistical production needs a unified combination of statistics and computer science in its very principles"*.

Motivated by the consensus that processes – in particular statistical processes **[2]** – for data-driven policy should be transparent **[3]**, we require software development and deployment in a statistical organisation to be open, reusable verifiable, reproducible, and collaborative. Beyond just devising guidelines and best practices, we show how the platform is implemented for the production of social statistics. For that purpose, we adopt a reasonable mix of *bottom-up* (from low-level scope to high-level vision) and *top-down* (from black-box process models to traceable functional modules) designs, so as to *"think global, [and] act local"* **[4]**. In building the parts while planning the whole, we provide with a flexible and agile approach to immediate needs and current legacy issues, as well as long-term problems and potential future requirements in statistical production **[5]**.

# Context

The development and deployment of a production system is not a trivial task. It can be successful only if data is managed properly through its life-cycle, by setting the right context to perform well-designed operations integrated into the right processes - *e.g.*, from collection, to dissemination through validation, integration, exploration and analysis - all in a timely fashion. Several challenging issues which accompany the practical implementation require specific focus *e.g.* in modelling and programming, in handling and managing data, in running the execution of processes, to mention a few. A production system needs to reach certain standards in terms of: *interoperability*, particularly when considering the mix of structured and unstructured data, qualitative and quantitative data; *accessibility*, since continuous operations may need to be achieved with no human intervention; *effectiveness and robustness*, to enable computational tasks to run effectively and consistently on data; *scalability and timeliness*, so as to provide quality statistics quickly and efficiently; *flexibility and evolvability*, because implementation usually happens without basic knowledge of the runtime targets. On that latter aspect, the maintenance and extension of a production system may reveal burdensome due to legacy issues, *e.g.* software/hardware components have changed or have become obsolete.

Altogether, and apart from other technical and methodological factors, a statistical production chain is a complex system **[3]**. It is usually difficult to describe production tasks in a homogeneous fashion since there is often a large number of intricate components and ambiguous operations. As a result, a human factor often adds to the complexity since inaccurate and out-of-date information may hinder efficient coordination and communication between the actors of the production chain – designers, implementers, and maintainers.

# Adopted design principles

Some practical requirements should be regarded as sound principles to gear the development of IT systems (*e.g.,* used in the production of social indicators) in a public statistical organisation so as to ensure complete transparency. *Openness* is one crucial aspect, so as to favour reusability and reproducibility **[6]**. It shall not be however intended as in *"open source software"* only, but rather as in *"open source code"* instead. Open source software solutions are obviously preferred **[7]** – since it guarantees openness and adaptability, though still susceptible to downsides – however the actual implementation is often tied to legacy proprietary software in prominent and wide use in statistical offices **[8]**. *Reproducibility* is an attainable minimum standard to ensure that the processes and computational workflows can be replicated. Another important requirement for integrated components is their *reusability*, *i.e.,* they should be applicable in contexts/domains other than the ones for which they have been developed. Again, a critical barrier is, in many cases, that the source code is not available. *Verifiability* is also desirable: given any model and set of parameters, it should be possible not only to regenerate the output of the production chain, but also to fully test and validate the underlying methodology. Full *in-house control* is a basic requirement that favours the continuity of operations, but also supports their future evolution. In principle, no actor involved in the production chain should turn into a bottleneck, *e.g.,* because some expertise is drained off or because the software system does not keep track of producers' actions in concrete form. Additionally, a strong control of the practical and effective implementation of statistical methods and techniques is also needed. Statistical software packages – may they be open or commercial-off-the-shelf –generally implement a broad range of techniques but do so in an *ad-hoc* fashion, leaving users at a disadvantage since they may not understand all the implications of a given analysis or how to test the validity of results. Finally, *sharing and collaboration* need to be supported: computational resources facilitate in-house participation and integration of additional contributions. Any skilled person could modify the code to suit his/her needs, learn from its use and further contribute to its improvement. This way, it can help adopting new efficient ways of working in production and collaborating beyond.

# practical implementation framework and current status

From a top-down perspective, it is useful to raise the level of abstraction when designing the production workflow, so as to consider generic processes instead of looking at single tasks **[5]**. The whole computational cycle can indeed be organised into processes following the classification framework provided by the Generic Statistical Business Process Model (GSBPM, see Figure 1). Raising the level of abstraction is also essential for enabling flexibility and evolvability since a production system will most likely change during its lifetime. First, this approach makes possible both vertical and horizontal integration by possibly adding up already existing and/or new processes/operations corresponding to different domains of the GSBPM – ideally, provided in a service-oriented manner **[9]**. Second, by focussing on overall processes instead of single tasks, it further releases the constraint on the choice of a programming language: it should be possible for the producer to implement a given statistical process in whatever programming language that fits best. Obviously, this *"language-agnostic"* approachhas a cost since it introduces further constraint on the implementation: it becomes necessary to ensure that the statistical output results obtained using different software are identical, *i.e.* that processes are robust **[10]**. Still, it offers additional control since it further guarantees not only the validity of these results, but the methodological choices made for implementation as well – e.g., on the way a Gini index, a quantile are calculated, sampling is performed… It also supports software (and hardware) independence in case of future migration. Currently, both R and SAS developments are integrated into the platform. Finally, note that considerations on the testing of statistical software, and all related aspects of statistical software reliability, need also to be taken into account.

# 

Figure 1 - A snapshot of the user-friendly documentation of PING platform. The organisation matches that of GSBPM. In addition, when applicable, processes, sub-processes and operations are described through BPM graphical representation.

We further combine this process-centric approach with bottom-up considerations, *e.g.,* specific methodological and technological aspects of the processes are also taken into account **[1]**. In practice, it means that operations along the production chain are modularised to overall support the whole production cycle, but also to solve immediate ad-hoc issues. By starting small and building quickly through the software platform, and by gaining experience from its deployment, it is possible to foster collaborative effort coming from different domains (of social statistics, but not only). Indeed, it provides a set of positive incentives for useful modifications to be shared back to the benefit of the entire community through continuously implementing small changes into the platform. This way, users shall also become producers, say *produsers*, and potentially contribute to new experiments and help develop/deploy more advanced analysis methods in-house. For that purpose, the software platform needs to become *social*. Methodological concepts and computational aspects, processes and programs are all fully documented. Beyond being fully tested, they are also exemplified. In practice, we aim at complying with the rules: "for every result, keep track of how it was produced", and: "provide public access to scripts, runs, and results" enounced in **[12]**. This approach currently enables us to share and reuse code – using standard versioning system – while guidelines for developing, documenting and testing programs are made available for *produsers* to further participate. Altogether, it contributes to a more holistic and extensive approach to production systems in statistical offices.

# Future plans

In the future, we plan to explore ways of moving along the dimension of scalability while again maintaining the generality of the platform. We believe that progress will be greatly facilitated by sustained and strong interaction between all *produsers*, with expertise cutting across multiple domains. Further, interrelation between standards for business process model and information model will also be explored.

# References

[1] Salgado D. (2016): A modern vision of official statistical production ([online access](http://www.ine.es/ine/premio_gespana2016.pdf)).

[2] Sutherland W.J. *et al.* (2013): Policy: Twenty tips for interpreting scientific claims, *Nature*, 503:335-337, doi:[10.1038/503335a](https://dx.doi.org/10.1038/503335a).

[3] Bertot J.C. *et al.* (2010): Using ICTs to create a culture of transparency: E-government and social media as openness and anti-corruption tools for societies, *Government Information Quarterly*, 27(3):264-271, doi:[10.3233/IP-140328](http://dx.doi.org/10.3233/IP-140328).

[4] Lalor T. and Vale S. (2013): Modernising the production of official statistics, *Revista Internacional de Estadistica y Geografia*, 4(2):72-79 ([online access](Modernising%20the%20production%20of%20official%20statistics)).

[5] UN Economic Commission for Europe: Common Statistical Production Architecture, *version 1.5* ([online access](http://www1.unece.org/stat/platform/download/attachments/120128612/CSPA%20v1.5%20.doc?version=1&modificationDate=1454409979907&api=v2)), Generic Statistical Business Process Model*, version 5.0* ([online access](http://www1.unece.org/stat/platform/display/GSBPM/GSBPM+v5.0)), Generic Statistical Information Model, *version 1.1* ([online access](http://www1.unece.org/stat/platform/download/attachments/75563844/GSIM%20Communication%201_1.docx?version=1&modificationDate=1387785453879&api=v2)).

[6] Ince D.C. *et al.* (2012): The case for open computer programs, *Nature*, 482:485-488, doi:[10.1038/nature10836](https://dx.doi.org/10.1038/nature10836).

[7] Templ M. and Todorov V. (2016): The software environment R for official statistics and survey methodology, *Austrian Journal of Statistics*, 45:97–124, doi:[10.17713/ajs.v45i1.100](http://dx.doi.org/10.17713/ajs.v45i1.100).

[8] Billings T.E. and Bankston E. (2013): Bridging the gap between SAS applications developed by business units and conventional IT production, *Proc. SUGI* ([online access](http://support.sas.com/resources/papers/proceedings13/485-2013.pdf)).

[9] Bruno M. et al. (2016): CORE: a concrete implementation of the CSPA architecture, *Statistical Journal of the IAOS*, pp.1-6, doi:[10.3233/SJI-160977](http://dx.doi.org/10.3233/SJI-160977).

[10] Wilkinson L. (1994): Practical guidelines for testing statistical software, in *Computational Statistics*, Physica-Verlag ([online access](https://www.cs.uic.edu/~wilkinson/Publications/accuracy.pdf)).

[11] Fiorenza P. *et al.* (2013): Ten steps to leveraging analytics in the public sector, in *Unlocking the Power of Government Analytics*, pp.29-32 ([online access](http://govloop.com/blogs/4001-5000/4845-AnalyticsGuide2013_final.pdf)).

[\*] Efron B. and Hastie T. (2016): *Computer Age Statistical Inference - Algorithms, Evidence, and Data Science*, Cambridge University Press.

# 

Figure 2 - A simple statistical operation is made available to the user with consistent implementation in different programming languages.