United Nations ECE/CES/2023/22

Economic and Social Council

Distr.: General
2 June 2023

English only

Economic Commission for Europe

Conference of European Statisticians
Seventy-first plenary session
Geneva, 22–23 June 2023
Item 3 of the provisional agenda
Moving towards open-source technologies – strategic and managerial perspective

Open-source solutions at Statistics Poland

Prepared by Poland

Summary

The document presents the policy of using open source solutions at Statistics Poland, indicates selected examples of solutions based on open source, defines ways to share them with the open source community, and discusses possible use cases. It also presents emerging opportunities and potential threats to using open source solutions in official statistics.

The document is presented to the Conference of European Statisticians' session on "Moving towards open-source technologies – strategic and managerial perspective" for discussion.

I. Introduction – Openness Manifesto

1. Statistics Poland adapts to the new digital world using available technologies and data in an innovative, safe, and affordable way. Considering the benefits obtained by society and the economy resulting from the use of open-source software, as well as the idea of Open Data, Open Algorithms, Open Access, and Open Knowledge, especially in the field of free access to scientific content, and also taking into account the law, and applicable rules and standards for statistical production, the President of Statistics Poland issued the "Being Open" Manifesto which declares:

- Use of open-source software in statistical products
- Active participation in the work of the community creating open-source software and sharing the results of their work
- Sharing developed codes to share experiences and improve the quality of created software
- Making statistical datasets available in open formats by open standards
- Increasing the usability and interoperability of shared statistical data sets
- Improving the content and functionality of shared statistical data sets, including access via API
- Sharing developed statistical data processing algorithms
- Creating and sharing open resources of statistical knowledge
- Development of tools for the use and sharing of open knowledge resources
- Active participation in the development of communities promoting and disseminating the idea of sharing open knowledge resources
- Popularization of scientific publications for the promotion of knowledge and the development of science in statistics.

2. Based on the "Being Open" Manifesto, Statistics Poland declares using open-source software in the statistical production process and active participation in the work of communities creating open-source software and sharing the results of their work. Implementing the above declarations is based on using and developing existing open-source products and creating and sharing solutions, building a community around them.

3. Using open-source solutions brings many benefits to projects implemented by units of Statistics Poland – it can affect the quality of software, reduce the cost of its development, and can also contribute towards lowering the barrier to public-private cooperation. Free and open-source (FOSS) solutions are an inherently transparent, participatory, and collaborative approach, revolutionizing how software is created, improved and used fully following the concept of Open Government.

4. The actions initiated by Statistics Poland bring measurable benefits and confirm the validity of the adopted direction and should be maintained.

II. Open source in Statistics Poland

5. Open-source software is used in almost all areas – from technical infrastructure (operating systems, server, and network monitoring), through programming, data processing and analysis, data sharing, project management, geospatial information, and Internet application security, to minor utility programs, supporting daily work.

6. Below are examples of usage of open-source software in selected areas by Statistics Poland:

(a) Infrastructure

- **Docker** – open-source software used to implement virtualization at the operating system level (called "containerization") to create, deploy and run distributed applications.

(b) Database

- **PostgreSQL** – relational database management system used in the Information Portal.
- **Apache Cassandra** – distributed NoSQL database for exploited data in the internally developed system for data collection management and survey processing (CORstat).
- **Anorm** – a library that manages access to databases from Scala/Java code.

(c) Security

- **Kali Linux** – Linux distribution designed for penetration testing. It contains tools for monitoring network traffic, collecting information about systems, networks, system configurations, password cracking, and vulnerability.
- **Owasp ZAP** – an automatic and manual security test tool – web application scanner.
- **Sqlmap** – a script used to automate processes related to searching for SQL Injection vulnerabilities in web applications.
- **Gpg4win** – a set of data encryption tools.
- **VeraCrypt** – a program for encrypting data on computer hard drives (entire drives or individual partitions).
- **Tau Argus** – a program that helps to impose statistical secrecy.

(d) Programming and project management

- **NetBeans IDE** – an integrated development environment (IDE) for Java, the main goal of which is to accelerate the construction of Java applications, including web services and mobile applications.
- **Eclipse** – integrated development environment (IDE) for Java, PHP, and other languages.
- **SharpDevelop** – development environment for creating projects in C#, VB.NET languages, and Boo.
- **Microsoft R Open** – Microsoft's R runtime distribution.
- **R-Studio** – is used primarily to compare data sets and create tables, charts, and summaries when analyzing data; a good solution for large databases, as it allows you to perform many operations quickly. The advantage is the ease of reading CSV, XLS, and XLSX files.
- **Shiny** – a development environment for the R language for creating web applications, used for testing data analysis and creating interactive charts.
- **Redmine** – managing projects and related subprojects and tracking issues. Additionally, it has forums, etc., used as a successor to JIRA.
- **NodeJS** – an open-source, cross-platform runtime environment for creating server-side applications written in JavaScript.
- **Visual Studio Code** – a source code editor supporting syntax of such languages as C++, C#, CSS, Dart, Dockerfile, F#, Go, HTML, Java,

JavaScript, JSON, Julia, Less, Markdown, PHP, PowerShell, Python, R, Rust, SCSS, T-SQL or TypeScript.

- **Python** - Python language environment used to write scripts related to Big Data analysis.
- **Drupal** - CMS system.
- **OpenJDK** – available and open-source implementation of the Java programming language.
- **Anaconda Navigator** – a programming environment for Python in data science.
- **Apache NetBeans** – development environment for Java.

(e) Version control systems

- **GitLab –** version control system and software supporting project management based on Git.
- **Git** – version control system.
- **TortoiseSVN** – a set of tools to access the SVN version control system from the file manager.
- **SVN** – version control system.

(f) Tests

- **SOAP UI** – software for testing applications using the SOAP (Simple Object Access Protocol) protocol consisting in exchanging information between applications using WebServices communicating via messages in the XML format. Supports functional, load, and security tests.
- **JMeter** – an application that enables manual and automatic performance tests of web applications.

(g) Projects

- **Archi – Open-Source ArchiMate Modelling** – a tool for modeling corporate architecture.

(h) Graphics

- **Inkscape** – a program for editing vector graphics (e.g., related to graphical data visualization in a range not available in MS Excel).
- **Gimp** – a tool for processing digital graphics and photos. It is used to process files posted on intranet sites.

(i) Geospatial information

- **QGIS** – geoinformation software – creating maps, performing spatial analysis, supports many functions and formats: vector, raster, and database.

(j) Auxiliary tools

- **Libre Office** – office suite.
- **NotePad++** – an extensive text editor – a replacement for the system Notepad. Polish interface, syntax highlighting for many programming languages.
- **WinMerge** – comparing contents of folders and files (e.g., Word).
- **7-zip** – file and folder archiver.
- **FreeFileSynch –** folder comparison and synchronization software.

- **Audacity** – a program for recording, analyzing, and non-linear audio editing with support for multiple tracks and the ability to import MIDI.
- **FileZilla** – managing files on disks and FTP servers.
- **LaTeX** – software for automated typesetting.
- **Color Contrast Analyser** – a tool used to check the digital availability of documents.

7. The above examples of open-source software complement the other tools used in Statistics Poland. For example, Zabbix, a system operating on LINUX Debian 10 platform, is successfully applied in infrastructure to monitor networks, servers, virtual machines, and applications. As a tool for monitoring, collecting, and analyzing data, Zabbix is very useful for network administrators, system administrators, and SOC operators. In the area of security, the key software used during penetration tests of web applications is OWASP Zap (an automatic scanner), sqlmap (a script for finding SQL Injection vulnerabilities), and Kali (a Linux distribution for conducting penetration tests). In the area of data analysis, there is increasing use of the R system (R Studio, R Open) and Python – used, e.g., for machine learning (the sckit-learn library) and text processing (the Pandas library). In front-end applications built to share data, well-known and popular frameworks are used, such as React.js, vue.js, and Node.js., CMS Joomla and Drupal systems.

8. These are just a few examples, but they show the growing popularity of this software in Statistics Poland – due to its flexibility, which means the ability to customize solutions to your needs, covering functional gaps and cost-effectiveness.

9. There are also areas where open-source software is even leading in a given field. One can distinguish areas of data science work where a standard technology stack and a global standard are open-source tools and an environment for developers where commercial tools could also be acceptable.

10. As part of the Data Science Academy created in Statistics Poland, particular emphasis will be placed on using software such as R, Spark, Python, Hadoop, etc. Obviously, there is a need for training for employees and raising their competence in this area. This approach allows us to build a group of people with appropriate competencies within the organization, able to share their experiences both within the organization and on the international community forum.

11. Another area is web scraping – a dedicated environment has been created at the Central Statistical Office for automated data collection from websites. Both ready-made open-source software (e.g., Scrappy) and own – scripts or programs written in programming environments (e.g., Python, C#) are used in this environment.

12. In Statistics Poland, steps have been taken to release software for an open-source community. One of the places where we share our solutions is the GitHub platform (https://github.com/statisticspoland), where a repository has been created, which contains, among other things, codes developed for the European Big Data Hackathon 2019 and a reporting platform for the Sustainable Development Goals. The R “BDL” package is also available – an overlay for the Local Data Bank API and a library for validating VTL rules. These constantly updated projects represent the first significant step toward creating an Open Repository of Statistics Poland.

13. In addition, in public contracts for software prepared for Statistics Poland by other commercial companies, where source codes are also purchased, we secure the ownership rights to these codes. Thanks to it, we become the owners of the source codes, and its further development as open source is possible. When a software version reaches an appropriate level of maturity, it can be published by us and made available for other use by the data science community.

III. Strategy for implementing open-source solutions

14. Based on the experiences to date, Statistics Poland is preparing a strategy for implementing open-source solutions, adapted to its needs, and the goals set for its statistics. Considering all the advantages and possible disadvantages of using these solutions, the law, and the standards of the statistical production process, a development concept in this area is being developed, which as a result, will be translated into an implementation plan.

15. We define the basic assumptions of using open-source solutions, good practices related to this area, and the principles we want to follow while implementing and developing open-source software. The first step should be to define the need to be met. This should be followed by analyzing the available open-source software that meets our needs, firstly focusing on the most popular and reliable proven repositories, like GitHub and SourceForge. An important element of the solution implementation process is the assessment of software quality and the evaluation of the support that the software has. An active community of users and developers around the software is a guarantee of stability of technical support. It is also essential to check how the software is documented. The next and crucial stage is reviewing and analyzing the licenses on which the software is based.

16. Implementation is only possible after testing any software solution, and the same principle applies also to the implementation of open-source software. Often, the implementation must be preceded by the need to adapt to the existing infrastructure or integration with existing solutions. Such solutions, like all others, require training for people who will work with them. Bearing in mind the need to ensure smooth work with the implemented solution, support for employees should be well-planned and organized – to help them solve emerging problems on an ongoing basis. Implementation and transfer of software to the organization does not release us from responsibility for them. During the entire life of the software, it is necessary to monitor its development, track the appearance of new versions and carry out updates of the software used. These are the key issues of our concept in using open-source solutions.

17. Nevertheless, we also want to promote the active participation of our employees in communities gathered around open-source solutions, encouraging them to share the source codes of their own solutions and solutions ordered from external contractors. Our concept of active inclusion in the open-source community also assumes that when ordering software from a third-party contractor, we take care to ensure ownership of the source code and appropriate licenses, enabling us not only to develop this software on our own but also to make it available in code repositories. In this way, we can not only rely on our capabilities to develop such software, but also benefit from the cooperation and support provided by the community formed around free software.

18. At this point, it is also worth pointing out that using open-source software in every case is impossible. We often do not use it for the most essential and critical purposes. open-source solutions are very extensively trying to encapsulate and support meaningful options. Looking through the prism of such cases, it should be noted that to use open-source in Statistics Poland, it is first necessary to reliably identify the needs – whether the given software meets the required functionalities and limitations. Specific requirements that this software does not meet may make its adaptation much more expensive than building a new solution from scratch, using internal resources, or using the services of an external contractor. Only such an analysis can make it possible to decide whether to use open-source software.

19. The larger the community around a particular solution, the greater the chance for continuous development and the greater the possibility of support. For this reason, switching to open-source software at any cost just to become independent of commercial solutions is impossible. However, where these needs overlap, open-source software can be successfully used. In such cases, an analysis of sustainability and the possibility of continued development is essential.

IV. Opportunities and threats for statistics related to the use of open source

A. Opportunities

20. An unquestionable advantage of using open-source solutions is the community's support in developing such software, especially in cases where this community is large enough. In this case, you can count on the support of high-class specialists who use these solutions daily. Often, such a community exchanges experiences on dedicated internet forums, so one can count on a quick response to emerging gaps and push software development in the right direction. This gives open-source solutions an advantage over commercial solutions.

21. Compared to commercial solutions, open-source solutions do not generate excessive costs. This is particularly important in the case of large organizations such as Statistics Poland, where the order or purchase of an appropriate number of licenses, even for inexpensive commercial software, is a significant cost in their budget.

22. Open-source solutions built with a large community of people interested in their development are also characterized by high durability, supported by frequent updates in identified gaps and faults. The sheer number of people using the software ensures it has adequate reliability and quick response to emerging problems without generating costs associated with providing sufficient support.

B. Threats

23. The decision to use open-source software should be preceded by an analysis of its popularity and the extent of support by the community. It should be remembered that using open-source software also carries risk elements. This is especially true when the community interested in a particular solution is small. In this case, features placed on the side of advantages may automatically become a threat. It is then essential to react adequately and quickly to potentially emerging threats.

24. In some cases, particularly those requiring a sufficiently high level of information security in the organization, using open-source solutions requires a very detailed analysis of the potential risk. It may also turn out that for critical tasks, open-source software cannot be used at all.

25. Another threat is the possibility of backward incompatibility or abandonment of open-source software development altogether, posing a risk to maintaining a given project. This includes the lack of further system updates and the need to migrate CentOS-based programs to other Linux distributions. During such a migration, problems may arise – software incompatibility and, therefore, the need to modify the application or individual modules. A good example is the CentOS system – a Linux distribution based on the Red Hat source code, where the project was completed in December 2020.

26. A significant threat to security is the community's lack of thorough verification of system components. One of the favorite attack points of hackers is plugins of all kinds – easy extensibility, which is often the most significant advantage of open-source software and can become a potential source of threat. Publishing harmful software, plug-ins, and templates or using outdated software without critical security patches can be a possible point of attack. As a result, computers are infected with ransomware, data leaks, IT infrastructure paralysis, etc.

IV. Conclusions and recommendations

27. The implementation of open-source solutions should be preceded by the development of a single strategy, taking into account the goals and needs of the organization, expected benefits, and outlining the areas in which we will implement these solutions.

28. When starting to implement a specific solution, it is necessary to analyze and evaluate various solutions and choose the one that best meets our expectations while ensuring adequate security for using this solution.

29. Open-source software should also be evaluated in the context of security. Security is not only testing or taking care of software updates. Since there are ready-made solutions in the form of programming libraries, it also poses threats related to the fact that libraries may cease to be developed in the long term. One should answer the following questions:

- How to effectively support further development of open source in the context of security?
- How to properly back-up libraries and protect against ones we cannot replace later?
- How should employees' competencies be developed to maintain continuity and further (also independently) develop software that cannot be replaced and no longer has proper open-source support?

30. Active participation in open-source communities means using their openness and readiness to help, which allows gaining knowledge and experience, learning from experienced programmers and experts. Participation in the community allows you to influence the development of the software that we want to use or use the quality and functionality of this software by reporting errors, code corrections, and suggestions for new functionalities. It allows us to build contacts, including international ones, and exchanging knowledge.