Three Levels of R Language Involvement in Global Monitoring Plan Warehouse Architecture

Jiří Kalina, Richard Hůlek, Jana Borůvkova, Jiří Jarkovský, Jana Klánová, Ladislav Dušek

Three Levels of R Language Involvement in Global Monitoring Plan Warehouse Architecture

Jiří Kalina[1,2], Richard Hůlek[1], Jana Borůvkova[2], Jiří Jarkovský[1], Jana Klánová[2] and Ladislav Dušek [1]

[1]Institute of Biostatistics and Analyses, Kamenice 126/3, 625 00 Brno, Czech Republic
[2]Research Centre for Toxic Compounds in the Environment, Kamenice 753/5, 625 00 Brno, Czech Republic

{kalina@mail.muni.cz, hulek@iba.muni.cz, boruvkova@recetox.muni.cz, jarkovsky@iba.muni.cz, klanova@recetox.muni.cz, dusek@iba.muni.cz}

Abstract. Three different options for involving R statistical software in the infrastructure of the data warehouse and visualization tool of the Global Monitoring Plan for persistent organic pollutants are presented, all differing in their demands with respect to data transfer rates, numbers of concurrently connected users, total amounts of data transferred, and the possibilities of repeating statistical calculations within a short period. After the development stage, two of these options were used at different levels of the system, demonstrating the specificity of their use and enabling the deployment of the powerful features of R statistical software by a system created using conventional programming languages.

Keywords: R · JSON · JSONIO · jsonlite · ODBC · statistical computing · web application · system architecture · POPs · GMP

1 Introduction

The ongoing collection of data on environmental pollution within the second campaign of the Global Monitoring Plan (GMP) concerning persistent organic pollutants (POPs) is benefiting from the experience gained during the first collection period (2003–2008) [1], which allowed the design and development of a system for the collection, statistical evaluation and visualization of data in order to achieve maximal efficiency of statistical processing while maintaining the highest possible information value hidden in the data.

The Global Monitoring Plan was established to provide the comparable monitoring of POPs listed in annexes of the Stockholm Convention (SC), i.e. their presence in the environment as well as their regional and global environmental transport. GMP implements this in the form of a worldwide overview of compliance by the monitoring and evaluation of SC 22 POPs concentration levels and their trends.

adfa, p. 1, 2011.

The data reporting model suggested in the updated Guidance involves compiling and archiving primary GMP data within a "regional data repository" in each of the five UN regional groups. In addition, regional data centers and a single GMP "data warehouse" should be established to compile and archive aggregated data, data products and results, including supplementary data that would be used in the Stockholm Convention effectiveness evaluation.

Based on the facts stated above, a multi-modular, on-line working data warehouse has been developed for data collection, processing and reporting within the second and future GMP collection campaigns. It is based on fully parametric data sheets for data input, supplemented by a possibility to utilize all data contained within the Global ENvironmental ASsessment Information System (GENASIS) repository [2]. It has been designed especially to improve the quality of the collected global data sets on POPs concentrations in order to determine their fate in the environment.

2 Architecture of R Involvement in the GMP Data Repository

The comparability of reported data is one of the elementary principles of GMP and also an essential condition enabling the use of a single set of methods for processing truly global data from all five United Nations Environmental Program (UNEP) regions, collected by dozens of individual (national) providers. The GMP principles of comparability became the elementary requirement for the development of the second generation of data collection and analysis tools. This comparability consists not only in the unification of all code lists used in reporting, but also in the standardization of physical units, the unification of the handling of missing values, the mutual conversion of the results of different methods of measurement and the fractions of substances measured, and the unification of the period for reporting, which was established during the previous decade with a one-year period.

The principal requirement imposed on the system for the collection and visualization of GMP data is the possibility of the simultaneous input of data either by transfer from existing databases (especially the GENASIS), or by the manual import of primary and (annually) aggregated data. Based on the data type and their input into the warehouse, the data undergo different-length sections of a sequence of mathematical and statistical operations, culminating in final outputs and visualizations with high added information value.

Since implementation of the statistical operations performed is very complicated and practically impossible using standard main and database programming languages (PHP and SQL in case of GMP), the series of analyses following logically in sequence demands the use of these languages only as an interface to the server version of the powerful statistical software R [4].

During development, three options for the involvement of the R language in the environmental pollution data assessment infrastructure of the GMP data warehouse system were tested (the transfer of data between the data warehouse and the specialized R server as a text file and procedure calls of R language using the HTTP protocol; data transfer between servers using JavaScript Object Notation (JSON) and direct

access to the R procedures using Open Database Connectivity (ODBC)) [3]. The advantages and disadvantages of different approaches have been shown to be specific in relation to the purpose for which the R script is used and the amount of data that is transmitted. Thus, the result of the development was mainly the differentiation of methods of integrating scripts of the R language into the rest of the GMP data warehouse systems according to the level at which this involvement occurs.

On their way through the GMP data warehouse system from their input in the form of primary data to the resulting plots and summary statistics, the data go through three levels:

1. The lowest level (data input): input and transformation of primary data inside the Genasis repository. Primary data go through the following four sub-steps:
   (a) Conversion of values measured by passive air samplers to atmospheric concentrations of pollutants using the method developed in the framework of the Global Atmospheric Passive Sampling network (GAPS) and implemented as a separate recalculating R package (genasis) [5].
   (b) Calculation of the parameter sums, defined as the sum of a large number of different parameters, including the calculation of the detection/quantification limits (LoD/LoQ) (e.g. sum of indicator PCBs).
   (c) Calculation of toxic equivalents for selected groups of substances defined as the weighted sum of concentrations of substances measured at the same time in the same locality, and the treatment of censored values (LoD, LoQ).
   (d) Aggregation of separately measured fractions of identical analytes to the general fractions (e.g. gaseous phase and dust for air sampling), and the treatment of censored values (LoD, LoQ).
2. The testing of prerequisites for the aggregation of measured values over time to mutually comparable annual averages, the aggregation itself, and the treatment of censored values (LoD, LoQ).
3. Real time calculations of descriptive statistics and trends while browsing individual analyses on the website.

Primary data enters into this sequence at the lowest level (1), the aggregated data at level 2. The necessity to implement the R script language in order to execute the calculations exists over the entire sequence (i.e. in all three steps and the four sub-steps):

2.1 Data Input Level

A specialized R package “genasis-recalc” was invented for elementary operations on primary data on persistent pollution in the atmosphere, water, human blood, and milk. Specific demands at different levels of aggregation (over fractions of the environmental matrix, groups of similar compounds and/or sets of toxic equivalent compounds, according to different toxic equivalents and factors schemes (TEQs & TEFs) of the World Health Organization), the substitution of left-censored data (LoD, LoQ) [6], and temperature-dependent coefficients of recalculations of passive air sampling were adopted inside the package and are used at the moment when the data enter the repository.

In this level, data transfer was implemented using txt files. Because the backend consisting of an R-server for statistical computing is, for safety reasons and for reasons of achieving better performance, separated from the DWH server as well as from the primary data source server GENASIS, the task of data transmission has become a key problem in terms of speed and stability. The numbers of individual records in primary databases range from thousands to millions, while each record contains tens of numerical and text values describing, besides the measured concentrations, also the site and date of measurement, the method used for the determination and recalculation of the values, the parameters of the measuring devices, data ownership, and the id of the measured substance and its chemical specification etc.

Recalculations of primary data are triggered manually by a data manager in logically defined batches containing up to tens of thousands of records, which is the equivalent of up to one million transferred numerical and text values. For building service-oriented infrastructure, the only viable solution, one necessitated by the conditions, is to use the text HTTP protocol, which raises the need to wrap different data structures (single values, vectors or numerical matrices) on the one hand, and expand them somehow on the other.

As a suitable language for both processes, the JSON notation for writing data structures was adopted, which was already used in the system. It is also involved within the relatively large OpenCPU computational platform used for the development of DWH. The JSONIO package was used on the R-server side, which, however, proved to be extremely memory-consuming for the order of thousands of records – on a server with limited memory, this approach led to overflow even at about 200,000 integer values and the increase in memory consumption appeared to be exponential.

The limiting factor was, therefore, not the complexity of the recalculations of the records on the chemicals themselves, but the coding of data into JSON notation, requiring extreme memory infrastructure regardless of the form of the extracted data (for the stated number of tens of thousands of entries, overflow occurred even in the case of a single vector, which itself would be insufficient for the transmission of real data).

A solution to extreme memory consumption was found by not using the JSONIO package but sending plain txt files with numerical and text values separated by a simple separator (a comma). It was, however, necessary to resolve the coding of complex structures (such as nested lists, matrices, etc.). In any event, such an approach solved the problem of memory consumption by large volumes of data. On the R-server side, the data were loaded from the file issued on the main language side using the scan() function, which excludes the use of the memory-consuming JSONIO package. The problem which arises from this approach is the need for the complicated coding of a more complex data structure to a simple text notation, which leads to inefficiency, instability, and a high frequency of errors. Furthermore, the time required by this approach is also relatively high.

As the most appropriate solution, the option of giving the R language direct access to the database was adopted, enabled by the RODBC R language package version 1.3-9 [7], published in late 2013 and implementing ODBC database connectivity. All R scripts are strictly separated from the GMP warehouse main language layer (there are

no direct data transfers between the main language and R scripts) and involved in the system using ODBC.

This form has proven to be the most effective in terms of memory consumption and acceptable in terms of time consumption. Because the four recalculations are performed on the primary data only once, and following steps work only with the results, the time consumption of the algorithm was not a crucial parameter.

The versatility of the ODBC connection also significantly increases the efficiency of the database solution from the perspective of experts from non-IT disciplines (especially environmental chemists and statisticians), who may develop computational algorithms in R completely separately and independently of IT developers and with only a limited knowledge of the language SQL.

The implementation of primary data recalculations was then performed in the form of a pair of buffers, represented by database tables (a properly constructed DB view and table) within the GMP DWH, between which the four resulting computations run according to the following cyclic procedure:

1. The data manager selects a batch (containing at most tens of thousands of records) for recalculation and, using a data management application, starts the filling of the input buffer (DB view), which is performed in the main programming language layer in DWH.
2. Directly from SQL, using the EXEC command, the appropriate procedure is called by R language on the R-server. Deploying the RODBC package function sqlQuery(), all necessary data is loaded into the R environment in the form of a data.frame structure.
3. If the loaded data.frame passes through a check of data completeness, it is recalculated and the results are stored, using RODBC again, in the output buffer (plain database table).
4. Still using the RODBC package and EXEC command, a follow-up procedure is called in the SQL database language. It is performed comprehensively in the main language layer of DWH, in addition to other necessary database operations, and finally the result is written back to the primary data structures.

Since the recalculations are performed in a predefined order, the data iteratively pass through all four steps when repeating this procedure for each recalculation (the calculation of passive measurement results of air pollution, the calculation of sums and toxic equivalents, and the calculation of fractions).

If the RODBC option of data transfer is used, the algorithms are less time-consuming, with a typical duration of units of minutes for the recalculation of each batch consisting of less than 1 million numerical or text values.

2.2 Aggregation Level

Once the data are recalculated and transformed to comparable scales, a selection of representative samples is carried out. In this step, the regularity of individual measurements over time (within one year) and the number of primary data are assessed

and, taking into account local specific conditions of monitoring, the representativeness of the primary data is checked.

At least 3 samples within one year and a form of monitoring regularity in which the longest gap between subsequent measurements is no more than 3 times longer than the smallest one are required in order for data to pass automatically through the selection process; in other cases, a manual check of representativeness is required.

Appropriate selected datasets are subsequently transformed to annual aggregations, which will culminate in achieving the maximal comparability of data from different data sources.

Since there are several specific aggregation functions within the process of annual aggregation, the R with ODBC connection to the DWH database was again used to avoid the complicated and unreliable constructions of computing values by means of SQL.

As in the previous case, the RODBC package is used for inputting and outputting R script data included in another special package "genasis-aggr". In addition to the usual statistics such as arithmetic and geometric mean and standard deviation, also maximum and minimum concentration values in each year, number of primary records, and several advanced statistics are computed:

- 5^{th} and 95^{th} percentiles as concentration values taken at regular intervals from the inverse of the estimated cumulative distribution function of the values within each year, which serves as a description of data variability without extreme values,
- the smallest, typical and largest gap between the start of the year, dates of subsequent measurements, and the end of the year, used for determining the regularity of the measurements,
- the number of values below detection/quantification limits.

These statistics cannot be computed directly by SQL without creating complicated procedures involving the complicated maintenance of special cases such as incomplete data records, low numbers of records etc., resulting in the low reliability of such a solution.

The use of direct access to the DWH database and the utilization of SQL commands within the R environment enable the determination of the optimal distribution of aggregation tasks between elementary steps, conducted with higher effectiveness by means of standard SQL commands and more specific computations.

2.3 Visualization Level

Another principle of R script involvement is implemented at the highest level of data visualization. The short response times and large number of multi-user operations associated with the use of a web-based data browser require a more flexible solution compared to working with ODBC data inputs and outputs. Moreover, a typical selection of parameters and localities for visualization inside a browser is highly restrictive and limits the amount of data to smaller values of hundreds of records, which can be rapidly transferred by the HTTP protocol between the components for processing individual tasks.

Another R package called “datavis-platform-gmp-r” was invented for this level of the visualization tool. JSON notation turned out to be the best transfer option for relatively small numbers of records. Both the input and output data of the R scripts used for visualization are translated to JSON notation using appropriate libraries (the jsonlite package on the side of the R-server is used [8]) and sent by the POST method of the HTTP protocol.

There are two visualizations where the use of R scripts was necessary due to complicated computations which cannot be implemented in either the SQL or PHP languages:

1. horizontal box and whisker plots of the descriptive statistics of selected measurements with different measures of central tendency and data variance (similar statistics as in the aggregation step are computed here using standard R statistical commands),
2. time series plots including linear and exponential interlacing curves and their confidence intervals, using several parametric statistical techniques not present in main level programming languages.

A data structure of nested data lists is the most appropriate for both these visualizations. This structure is highly suitable for translation to JSON by the jsonlite package, with the exception of singleton lists, which are translated as simple elements. This unsystematic property necessitates the special measure of adding one zero element to each level of the nested list to obtain a minimum length of two elements for each list.

3 Conclusion

There are different demands with respect to data transfer rates, numbers of concurrently connected users, total amounts of transferred data, and the possibilities of repeating statistical computations within a short period at different levels of the GMP data warehouse and visualization system. During the development of second generation GMP data browsers, three different options for involving R statistical software were tested and assessed for final use.

The method of using JavaScript Object Notation (JSON) for coding data transferred between a database running on a data warehouse server and a separate R-server for statistical computations was shown to be fast but unsuitable for larger amounts of data (the order of millions of values) due to its extreme memory-consuming properties (using the RJSONIO package in the R environment).

A different method of data transfer in the form of specially devised coding within txt files, which should have solved the problem, exhibited problems with more complicated data structures such as matrices and nested lists; in addition, it was also more time-consuming.

As the best option for huge amounts of data, direct access from the R-server to the data warehouse database was implemented, using the Open Database Connectivity (ODBC) standard, made available by means of the RODBC package. This solution allows SQL and R functions to be combined within one script and provides a simple,

effective and comfortable environment for the development of enviro-statistical structures separate from the rest of the IT infrastructure.

The main disadvantage of the ODBC approach, namely lower speeds of multiuser access to the database, was solved by using JSON whenever smaller numbers (in the order of hundreds) of values are transferred, such as in web-based visualizations.

References

1. Hůlek, R., Jarkovský, J.,Borůvková, J., Kalina, J.,Gregor, J., Šebková, K., Schwarz, D., Klánová, J., Dušek, L. *Global Monitoring Plan of the Stockholm Convention on Persistent Organic Pollutants: visualization and on-line analysis of data from the monitoring reports*. 2013.
2. Jarkovský J., Dušek L., Klánová J., Hůlek R., Šebková K., Borůvková J., Kalina J., Gregor J., Bednářová Z., Novák R., Šalko M., Hřebíček J., Holoubek I. Multi-matrix online data browser for environmental analysis and assessment [online]. Masaryk University, 2014. Version 1.0. Available from WWW: `www.genasis.cz`. Version 3.10. March 2014 [2014]
3. Hůlek, R., Kalina, J., Dušek, L., Jarkovský, J. Integration of R Statistical Environment into ICT Infrastructure of GMP and GENASIS. In Jiří Hřebíček, Gerald Schimak, Miroslav Kubásek, Andrea E. Rizzoli. *Environmental Software Systems : Fostering Information Sharing. IFIP AICT vol. 413*. Heidelberg: Springer, 2013. 240-252, 13 s. ISBN 978-3-642-41150-2.
4. The R Project for Statistical Computing. `http://www.r-project.org/` (2014)
5. Kalina, J., Klánová, J., Dušek, L., Harner, T., Borůvková, J., Jarkovský, J. *genasis: Global ENvironmental ASsessment Information System (GENASIS) computational tools. R package version 1.0.* 2014. http://CRAN.R-project.org/package=genasis.
6. Van den Berg, M., Birnbaum, L. S., Denison, M., De Vito, M., Farland, W., Feeley, M., Fiedler, H., Hakansson, H., Hanberg, A., Haws, L., Rose M., Safe, S., Schrenk, D., Tohyama, Ch., Tritscher, A., Tuomisto, J., Tysklind, M., Walker, N., Peterson, R. E. *The 2005 World Health Organization Reevaluation of Human and Mammalian Toxic Equivalency Factors for Dioxins and Dioxin-Like Compounds.* Toxicological Sciences 93 (2): 223-241. doi: 10.1093/toxsci/kfl055.
7. Ripley, B., Lapsley, M. *RODBC: ODBC Database Access. R package version 1.3-9. 2013. http://CRAN.R-project.org/package=RODBC.*
8. Ooms, J., Duncan, T. L., Hilaiel, L. *jsonlite: A Robust, High Performance JSON Parser and Generator for R. R package version 0.9.13.* http://cran.r-project.org/web/packages/jsonlite/index.html