Reproducible Data Processing, Aggregation, Reporting and Storing of Business Tendency Survey Data in an Open Source Environment

Matthias Bannert

August 16, 2013

Abstract

This paper describes a license cost free open source software environment[1] to implement reproducible and transparent handling of data stemming from business tendency surveys (BTS). Using R Studio Server as a centerpiece, data can be processed automatically using batches or handled interactively using the R programming language. Further this paper explains how to store the resulting panel data sets in a standard relational database as opposed to proprietary time series software. By centering around a scripting language that is used widely among social scientists transparency is increased dramatically at the relatively low costs of a loss in computing power as opposed to compiled general purpose language based work flows.

Keywords: business tendency surveys, cost cutting, reproducibility, transparency, aggregation, software architecture, reproducible research.

[1] The framework suggested in this paper is based on the R package introduced in Bannert (2013). This compact version was set up to fulfill the requirements of the EESW 2013 in Nuremberg.

1 Introduction

Very much like official statistics business tendency surveys are conducted on a regular basis. Thus there is the need for an environment that enables researchers to smoothly conduct surveys, quickly process, analyze and store data as well as to create standard periodical reports in relatively short time frame. Traditionally setting up such an environment has not only been costly, but it has been clearly assigned to IT personnel. In turn data processing has often not been entirely transparent to economic researchers who lack a background in generational purpose programming languages such as C or Java respectively have limited experience in database querying using different SQL dialects (or even non-relational databases). This paper suggests an environment that is completely free of license costs and is centered around the interactive programming language R (R Core Team, 2012). R is very popular among empirical researchers of almost any discipline and can improve transparency through the fact that is not compiled and common to researchers.

Reproducible Research

In recent years empirical economists in academic research have faced a growing pressure to make their results reproducible. To validate the results of data-driven research it is inevitable that referees can re-calculate results by running the source code and data used by the original author(s). What is known as `Reproducible Research` in the academic literature has ever been a requirement to more repetitive official statistics and business tendency survey reports.

Re-traceable Research

Yet the ability to re-calculate results elsewhere does not only require access to the source code but also to the original dataset. Obviously sharing data – particularly at the micro level – is often precarious due to privacy reasons. In turn this fact emphasizes the importance of meta data: If results cannot be fully reproduced proper description of the data generating process is indispensable as well as being able to track data down to its providers and analysts. Data processing of any kind needs to be designed to attach meta information generically while the analysis is conducted. In the case of business tendency survey data this means that fixed meta variables like data unit or originator as well as localized information such as question wording or answer items need to be attached to the data.

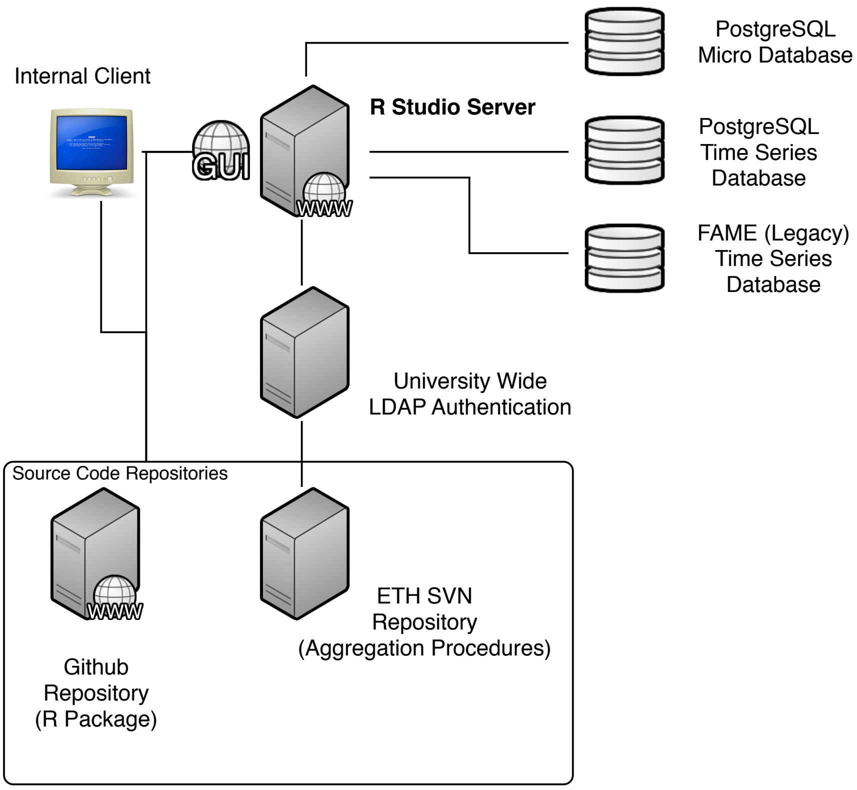
2 Implementation

Open source software has become increasingly powerful in virtually all fields that affect the process of maintaining and developing business tendency surveys. Though there are decent open source projects dealing with the online and offline conduction of surveys this paper focuses on storage and processing of business tendency surveys. The most common need in business tendency research is to aggregate survey data on company level to different kind of sector or regional levels. Though mathematically trivial, schemes can become quite complex due to weighting, sector classification and multi step aggregation.

Server Architecture

Thus a framework which favors collaborative development and documentation is required. Moreover separation of concerns such as generating[2], processing and storing the data needs to be guaranteed. Figure 1 suggests an architecture (Bannert, 2013) that fulfills these requirements and is entirely based on open source software.

Figure 1: Open Source Framework Centered Around R Studio Server



Data at participant level (i.e. raw survey data) as well as sector or regional weights are stored in a relational PostgreSQL micro data archive. Clients that are authenticated by an university wide LDAP[3] system can access R using the web based graphical user interface provided by R Studio Server. With the help of the R packages `RPostgreSQL`, `ROracle` and `fame` installed on the server, clients can use the server to access micro data as well as aggregated time series. Again access at the database level is negotiated with the university's Active Directory. Source code is stored in two different version controlled repositories: general libraries for the aggregation of survey data are bundled in the custom built R package `gateveys` (Bannert, 2012) which is publicly available

[2] I.e. conducting a survey with a third party tool which uses a transaction database. Though we have some experience with various versions of *lime survey* discussing the data generating process itself is beyond the scope of this paper.

[3] We use Kerberos to read from the university's Active Directory instead of using system users on the RHEL 6.3 machine that hosts our R Studio server.

from github.com/mbannert. Survey specific aggregation procedures which are basically function calls using functions of the gateveys package, are stored in an internal SVN repository. This architecture allows for centralized maintenance of database drivers and R package versions as well as R itself. Besides there is no need for any client software except a modern web browser that supports web sockets. The current setup can easily be scaled up or extended by other web interfaces.

The gateveys R Package

The name gateveys stands for aggregate surveys and describes the main purpose of the functions within this library. We make use of R's package architecture which favors context documentation of our libraries. Our bundle contains standalone examples for all of its major functions as well as a random generator to set up datasets that mimmick typical business tendency survey datasets. We also provide the most recent NOGA sector classification as an R dataset which ships with the package. Packaging makes our software easily transferable to other machines running on either Windows, Mac OS or Linux.

Aggregation

All aggregation is calculated on script level using un-compiled R scripts as opposed to compiled languages or database queries. Our aggregation procedures enable researchers to re-calculate an entire panel as opposed to adding new waves incrementally only. Thus newly introduced sector classifications or weighting schemes can easily be calculated for older data belatedly. By exporting a group of required packages we make sure users find and download all dependencies. We also limit required packages to popular and established packages which are available from CRAN[4]. Further we provide the opportunity to use parallel processing for some of our workhorse functions. Parallel processing is used automatically when the corresponding parallel versions of the functions are loaded as part of the optional R packages foreach (Revolution Analytics, 2012) and multicore (Urbanek, 2011).

Referencing and Object Mapping

Obviously the number of observations is reduced in the process of aggregating micro data to higher levels. Nevertheless various aggregation levels, regional aggregation or different weighting schemes result in specific time series objects per variable. Hence survey researchers end up with a plethora of variables at the aggregated levels: Thousands of time series objects need to be referenced and meta data needs to be attached in order to make data accessible for others.

Our aggregations create SDMX[5] inspired time series keys generically for each result. Besides two kinds of meta information are attached to every time series during the aggregation process:

[4] The Comprehensive R Archive Network is the standard resource for R packages which is available as a source for R extension from within R. CRAN packages have to fulfill certain criteria are maintained and supervised by package maintainers.

[5] Statistical Data and Meta Data Exchange. SDMX is an initiative to foster standards for the exchange of statistical information that is supported by BIS, ECB, EUROSTAT, IMF, OECD, UN and World Bank. See also http://www.sdmx.org.

fixed meta data as well as localized meta data. Fixed meta data refers to untranslated meta information such as units, other proprietary reference keys or item counts. Hence only one object of class `metaFixed` can be attached to each time series object. As opposed to that researchers can add one translated meta information object of class `metaLocalized` for every language to one time series object. Question wording or answer items are just two examples of localized meta information that help to make sense of the data or simply add labels to figures and tables.

Finally we store the resulting object along with its meta information into a relational database[6]. We extend an idea suggested by Paul Gilbert who came up with a mapping of R time series objects along with their meta information in to an relational database. His `TSdbi` (Gilbert, 2012) family provides an extra table for every frequency and a table for all meta information. We extend this model to multiple meta information tables for different languages. The `gateveys` package enables researchers to map R objects created by the aggregation process described before into database entries as well as the other way around.

3 Discussion and Outlook

The suggested framework seems to be well suited for the aggregation of business tendency survey data. Its main drawback, namely lower computational performance compared to compiled approaches, does not weigh in heavily in the case of business tendency surveys because they will not produce huge datasets respectively data will not grow at the speed of technological progress. Moreover performance concerns can be attenuated by making use of multiple processors as operations are well parallelize-able: questions do not depend on each other and can be calculated variable by variable. Also survey results only need to be published in a matter of days or even weeks sparing the issue of super timely calculations. Finally by using a language that is widely used and interactive, researchers can check intermediate results on the console and interact with their calculations.

All parts of the framework come free of license costs: Linux servers, PostgreSQL as a database, R as the main scripting language and R Studio as the central IDE. Further extensions are planned using Shiny web applications to provide interactive results to the end user.

[6] Basically any relational database that R can connect to can be used to store the data. We haven chosen PostgreSQL, but have also worked with MySQL and Oracle at the database level. Furthermore we have worked with Sunguard's non-relational time series database FAME to store result data. The latter can only store and organize meta information to a limited degree.

References

Bannert, M. (2012): "gateveys: An R Toolbox for Business Tendency Survey Researchers," https://github.com/mbannert/gateveys/, R package version 0.1.

——— (2013): "Gateveys –An R Toolbox for Re-traceable Aggregation of Business Tendency Survey Data," *KOF Working Papers*, 326.

Gilbert, P. (2012): *TSdbi: TSdbi (Time Series Database Interface),* R package version 2012.8-1.

R Core Team (2012): *R: A Language and Environment for Statistical Computing,*R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0.

Revolution Analytics (2012): *foreach: Foreach looping construct for R,*R package version 1.4.0.

Urbanek, S. (2011): *multicore: Parallel processing of R code on machines with multiple cores or CPUs,* R package version 0.1-7.