D-Lib Magazine

The Magazine of Digital Library Research

D-Lib Magazine

January/February 2017
Volume 23, Number 1/2
Table of Contents

Enabling Reproducibility for Small and Large Scale Research Data Sets

Stefan Pröll
SBA Research, Austria
sproell@sba-research.org

Andreas Rauber
Vienna University of Technology, Austria
rauber@ifs.tuwoen.ac.at

https://doi.org/10.1045/january2017-proell

Abstract

A large portion of scientific results is based on analysing and processing research data. In order for an eScience experiment to be reproducible, we need to able to identify precisely the data set which was used in a study. Considering evolving data sources this can be a challenge, as studies often use subsets which have been extracted from a potentially large parent data set. Exporting and storing subsets in multiple versions does not scale with large amounts of data sets. For tackling this challenge, the RDA Working Group on Data Citation has developed a framework and provides a set of recommendations, which allow identifying precise subsets of evolving data sources based on versioned data and timestamped queries. In this work, we describe how this method can be applied in small scale research data scenarios and how it can be implemented in large scale data facilities having access to sophisticated data infrastructure. We describe how the RDA approach improves the reproducibility of eScience experiments and we provide an overview of existing pilots and use cases in small and large scale settings.

Keywords: Reproducibility, Research Data Sets

1 Introduction

Scientific results are based upon analysing and processing research data. Many new disciplines such as computational biology, computational sociology or computational physics have emerged, which accommodate the influence of computational methods on their traditional field, by including the term computational in the discipline name.

The trend of digitizing research and utilising more tools, more data and more complex research infrastructures, increased the quantity of eScience experiments in the recent years. The bandwidth of data driven research spreads from little to big data. Researchers may work with relatively small data sets on their own laptops or desktop computers or rely on large research facilities with complex machinery providing dedicated infrastructures. In both worlds, the amount of research data is increasing, which also produces a larger management overhead of handling the data sets.

Recently the issue of reproducibility has gained a lot of attention. In general, an experiment is reproducible, if and only if consistent, scientific results can be obtained, by processing the same data with the same algorithms using the same tools. Currently, the paper based article is the prioritized form of communicating scientific results. Although the willingness to share data and processes is increasing [1], data is often cited insufficiently or not accessible, which hinders the reproducibility of experiments and

lowers the trust in research results.

On a high level perspective, for an experiment to be reproducible, we need to have knowledge of at least the following information:

- Research data and metadata used
- Methods applied in the experiment
- Tools, software and execution environment used in the experiment

For obvious reasons, how reproducibility can be achieved differs between disciplines. Also, the decision on when a result can be claimed as being reproducible is still open for discussion. The authors of [2] suggest to distinguish between methods reproducibility, results reproducibility and inferential reproducibility. Methods reproducibility covers experiments, where enough details are available for the exact reproduction of an experiment. Results reproducibility covers replicability, which is the repetition of an experiment as close as possible with the original experiment. Inferential reproducibility deals with the question, can the same conclusions can be made from an independent replication of an experiment. All three replication types form together the three dimensions of reproducibility. A different view of reproducibility and specifically the gains in terms of knowledge they provide, is being proposed in the PRIMAD model [3]. This model differentiates between the aspects of an experiment that can be modified (such as the data, platform, implementation, method, research objective and actors) and the benefits and insights we gain from it (e.g. robustness, generalizability, portability).

So far, reproducibility covered mainly the question, can a claim which has been made in a publication be verified. In theory, the verification is then achieved by retrieving the data and the software, which is referenced with textual descriptions or with links in the publication. Recently, these links are mapped by persistent identifiers (PIDs), which hinder link rot and provide access to resources in the long term. For the actual verification, researchers apply the described methods and if these are applied properly, the same results can be reproduced [4].

This entails that all information contained in the publication must be available and that the methods have been described with in-depth precision, which allows peers without any ambiguity to execute every step of an experiment and verify the outcome. While it is a challenge to reproduce the execution environment for running the verification process [5], obtaining the correct data set is an issue on which we are focussing in this work. In order to reproduce a result from a study or an experiment, researchers need means for accessing precisely the very same data again, as it was used in the original study or experiment. While the term of originality is hardly applicable to data due to the nature of copies, it is a clear requirement that a data set must be complete and unchanged, having precisely the same properties it had in the first experiment.

Researchers rarely use data sets as a whole, but rather extract subsets from large and evolving data sources. There exists a discrepancy between the requirement for keeping up with growing and changing data sets and with creating static and citable subsets thereof. Although it was suggested [6] that manual data manipulation steps should be avoided and raw data should be kept, it is clear that creating copies of each version neither scales with very large data sets, nor does it scale with large volumes, even of small data sets. Considering the dynamics of research data while providing access to a precisely defined, static subsets without relying on copies is the goal of our work. We investigate how data sets and subsets thereof can be made reproducible. We describe how precise data identification and data citation can contribute to the method reproducibility and the result reproducibility perspective of eScience experiments.

2 Research Data and Subsets

Research data is a fundamental building block of eScience experiments. For this reason, it is a requirement for reproducible experiments, that the data used is available. There exist several pre-requisites which must be met in order to provide the data to peers which can then verify experiment.

- Data must be available and accessible
- Data (i.e. the specific (sub)set) must be precisely identifiable
- Data must be available in the specific version used

With an increasing amount of data in terms of size and complexity, researchers hardly use the whole data set at once, but rather create subsets of a larger data set. By a subset, we understand a selection of records and a projection of properties of the data set, based on defined parameters. Subsets are often specific for one particular study and contain implicit domain knowledge. Unfortunately, researchers rarely describe or cite the subset used in a study in the paper they publish in sufficient level of detail for several reasons. First of all, describing research data is a tedious task and therefore often avoided, as it is easier (if done at all) to cite a single, large data set than individual subsets. Secondly, in some cases there does not exist proper standards or conventions for how a subset can be described properly so that it becomes citable for peers. Textual descriptions of specific subsets are often not precise enough. Thirdly, research data is often not static but rather constantly updated, as corrections and additions are being made to the data in the database. Exporting a new subset after each database update and — as a consequence — storing copies of all subsets of an evolving data source does not scale.

3 Precise Subsets from Evolving Data

In order to tackle these issues, we introduced a concept that utilises the fact that subsets are created by extracting a portion of a parent data set. The parent data set is often cited well, but the subset which is actually used is not. Often the parent data sets are evolving, which makes identifying the correct version of a subset a complex challenge. Instead of exporting subsets each time an update was made, we version the records in a database and utilise timestamped queries, which are used for retrieving the subset at a later point in time. As we store the timestamp when the query was executed, we can create a match between the database state and the query execution. The metadata required for this approach is stored in a so called query store [6]. The query store records for each query the timestamp of execution, the query string, a hash key of the result set for the verification and the significant properties of the subset. These are filters, record sequences, record sortings and any other properties which have an influence on which records are included or excluded from a subset. The query store serves as the central infrastructure, for creating subsets and retrieving them again at any point in time based on re-executing the query on versioned data. These principles have been elaborated in detail by the RDA Working Group on Dynamic Data Citation (WGDC) and have recently been adopted as official RDA recommendations [7, 8].

Many researchers from the smaller scale data realm (sometimes also referred to as the "long-tail") often do not have sophisticated database management systems in place in order to manage their data. Although our approach requires an interface to the data, which allows creating subsets by issuing queries, it is not limited to databases offering query languages. The same principles can be applied to other data stores such as file repositories, and data exposed as XML files as well. We can interpret any process which allows creating a subset in a deterministic way as a query, even the selection of files in a file system or the execution of a script to obtain a subset using, for instance, the statistical language R. The query store can be used to record all parameters required in order to recreate a subset again, by executing the script on versioned data in a file system in an automated fashion. By versioning the data and time-stamping the subsetting process, we can retrieve highly specific data sets on demand, also by using external tools for creating subsets. This renders our approach useful for small and large scale research data scenarios likewise [9].

4 Reproducible Subsets

The query based approach offers several advantages compared with the export of static files. Firstly, by attaching the persistent identifier to the query directly, researchers gain advanced features for working with subsets from evolving data sources. In contrast to static file exports, the query based approach allows not only retrieving precisely the same version of a subset again, but also to retrieve previous or newer versions of a data set with the same specifications. Researchers can track the changes in a data set and compare subsets from different points in time with each other.

Secondly, the query store captures valuable provenance information automatically. The query itself contains the parameters of a subset and the query store preserves additional metadata about authors, users, parent and derived datasets, hash values and execution date of a query and its result. We can parse and evaluate the query which created a subset automatically and extract metadata about a subset. This allows us to collect metadata about a subset and use this data for collecting provenance data about a subset. The query store as central instance keeps track of all identifiers and establishes and maintains relationships between them [8]. We can use the collected metadata for displaying details on a human readable landing page and we can also offer APIs for machine consumption. With this technique, it is feasible to find for instance all studies using subsets, which would be affected by an update in a parent data set. In addition, we can measure the usage of a data set and provide metrics about data. The query store can also provide citation text snippets, which can be copied and pasted into reports and reduces the burden on the researcher.

Thirdly, the query store creates a link between the query and the data set (for instance a table in a database). All subsets which have been created are identifiable with their unique PID and can be retrieved again, without having to store extra copies of the subsets permanently. Thus, the storage demand is much lower, as we only need to store the metadata of the query execution and the query itself. Versioning the data in the database does produce an overhead, but as database versioning is a common and good practice already, this drawback has little negative influence.

The same principles can be applied to small and large scale data and improve the understanding how a data set was created. They apply to data sets with high frequency changes, streaming data or static data likewise.

5 Little Science and Small Scale Data Sets

Proper data management depends on the allocation and availability of resources. While larger research institutions often have dedicated data management infrastructures and trained personnel available, smaller scale research does not have access to sophisticated data management facilities and often lack the processes, which are necessary for storing the data, providing support and keeping research data accessible [10]. Research data often resides on the local workstation of a scientist and is in many cases not shared or made available, as many barriers exist [1].

From a technical perspective, the versioning of local research subsets can be addressed for instance by using versioning approaches

utilised with success in the domain of software development, such as Git [11]. In addition, researchers need to share the process of subset generation, for instance by providing detailed information about the used operating system, the tools and their parameters. In many cases it is sufficient to include the script used for extracting a subset from the parent data set into the version system as well and share the repository with peers [9]. The script may contain comments as further metadata which enhances the reproducibility of the subsetting process. Such a repository bundles together all the information which is necessary for obtaining a subset of an evolving data set again, by using open source and freely available tools such as Git.

Minting and assigning persistent identifiers requires resources which are often also not available, for instance because the fees are too high for small institutions. Recently, initiatives such as Zenodo allow attaching DOIs (Digital Object Identifiers) to Github repositories as well, enhancing the unique identification of repositories and their citation in the scientific and academic realm. By attaching persistent identifiers to the repository and including links between the parent data and the query, the subsetting process can be made reproducible and does not rely on copying subsets as the parent dataset evolves. Each subset can be re-created as it was at a given point in time. As a result, scientists wanting to reproduce the results can identify the repository via its PID and then proceed creating the subset again by executing the versioned query/script on the likewise versioned data set based on the provided information.

It is clear that the issue with the lack of data management in small scale settings needs to be addressed on a social level. There is a need for encouraging transparency by data sharing also for little science, as currently we see that the barriers for creating citable subsets of evolving data sources are lowering, also for little science.

6 Large Scale Data

In contrast to little science we can see a professionalisation of data management services in the big data domains, where sophisticated infrastructure is available for scientists to store their research data. Although researchers can deposit and retrieve data sets based on defined processes, the issues of identifying and obtaining specific subsets remain. Subsets used in studies are often only described textually in papers and no individual identification mechanism is provided.

From a technical point of view, the data is stored in large and distributed databases. These services are operated by professional database administrators. In scenarios dealing with large data, researchers access data sets often not directly, but rather via interfaces or trained data custodians. Instead of executing queries themselves, researchers rather provide a specification of a sub-set or the parameters, which describe a subset based on facts, in a workbench environment. The actual subsetting process is transparently hidden and is thus not part of the publication.

This information can be stored in the query store, either automatically by intercepting the interface calls or by the data custodian, who deposits the selection criteria as additional information in the query store, including the original query used for obtaining the subset. In both cases, the information needed for creating a subset is stored persistently in the query store and can be used for obtaining a subset again. As the data sets are in many cases already versioned, creating a link between the information on how a subset was obtained and the correct version can be achieved by storing timestamps in the query store.

As large institutions often already have a contract with a PID provider, assigning PIDs to the queries in the query store hardly creates additional costs. Thus enabling precise data identification also for subsets can be established by integrating a query store into the existing research data landscape.

7 Use Cases and Real World Scenarios

In order to evaluate the feasibility of our approach in small and large scale research environments, we collected a set of use cases from small and large scale research data projects, which implemented the RDA WGDC recommendations for data citation:

- Reference Implementations for CSV and XML
- Reference Implementation for flat files
- TIMBUS Project (SQL)
- Biomedical Big Data (WUSTL, i2b2)
- Marine research data (BCO-DMO)
- Ecosystem monitoring data (VMC)
- Oceanographic data (Argo)
- Astronomy research data (VAMDC)

For evaluating smaller scale research settings and little science, we developed two reference implementations for CSV and XML data. For the XML prototype, we developed a versioning scheme for the XML files and implemented a query mechanism based on X-Path, which allows extracting versioned subtrees of an XML data set. Each change within an XML file is traced by adding timestamps on

record level. The query store itself is an additional XML file containing the metadata for selecting subtrees. For CSV data we implemented two prototypes. One approach utilises a migration approach, which automatically imports CSV files into a relational database management system and creates a relational table structure for each individual CSV data set. Versioning is achieved on record level by storing individual changes, deletes and inserts and their time of occurrence in a dedicated query store, which is also a relational database. Researchers can utilise a Web interface for selecting, filtering and sorting. The system automatically records all these operations and stores them in the query store, where a persistent identifier is assigned to the query. Upon re-execution, the system utilises the data stored in the query store and re-executes a normalised query on the versioned data. The second approach operates on CSV data which is stored in a Git repository. The prototype provides a simple SQL interface and stores the queries also in a dedicated branch in a Git, allowing the re-execution of the query on a specific version of a data file. This approach implements file level versioning.

In large scale scenarios we evaluated the feasibility of our approach within the TIMBUS project, which utilised a relational database for storing measurement data. Users would utilise a complex web interface for creating reports including graphs of the measurements within a specific time frame. The subsets which are created transparently in the background serve as the data source for the plots. By implementing a query store which captures these parameters, the process of creating the subsets for the graphs can be made reproducible. Each subset can be identified by a persistent identifier and this identifier can be included into the printed report, allowing not only to retrieve the graph but also the underlying raw data again.

Currently the RDA Recommendations for Data Citation are implemented at several sites into existing systems. The pilots which are currently active span several domains, from eHealth to oceanographic data to nuclear research data. In most scenarios the data was already versioned, but end users did not have access to previous versions of the data sets. The main effort in the implementation projects was integrate a query store which can capture the subsetting process and provide interfaces for retrieving specific versions of subsets again.

8 Conclusions

Utilising versioned data and timestamped queries provides a flexible way of creating precise and citable subsets of evolving datasets. By storing the details of a query, we can understand how a subset was created and utilise this knowledge in order to achieve reproducible subsets of data. As we require the subsetting process to be deterministic, the approach delivers the same results when applied on the same data. For this reason, the query based approach supports methods reproducibility, as it provides exactly the same data again and offers additional metadata about a subset, which can be used for verifying the integrity and completeness of a subset.

Based on the fact that our approach requires much less storage in comparison with exporting individual subsets, we can also utilise the queries to create subsets from different time periods, when the data is evolving. Thus we can evaluate how the subset changed in the course of time. The concept also allows adjusting the parameters in a controlled way, in order to get similar subsets and therefore allow study of the effect of small changes and test the robustness of results. Our approach also considers the technical advancement and it allows migrating data and queries into a different system. As the details about a query are known, we can translate between different storage systems, query languages and systems and verify if both systems still produce the same results. This flexibility supports the result reproducibility.

References

[1] Tenopir C., Dalton E.D., Allard S., Frame M., Pjesivac I., Birch B., *et al*. (2015) Changes in Data Sharing and Data Reuse Practices and Perceptions among Scientists Worldwide. *PLoS ONE* 10(8): e0134826. https://doi.org/10.1371/journal.pone.0134826

[2] Steven N. Goodman, Daniele Fanelli, John P. A. Ioannidis. What does research reproducibility mean? *Science Translational Medicine*. Vol. 8, Issue 341, pp. 341ps12. 2016. https://doi.org/10.1126/scitranslmed.aaf5027

[3] Juliana Freire, Norbert Fuhr, Andreas Rauber. Reproducibility of Data-Oriented Experiments in eScience. *Dagstuhl Reports*, 6(1), 2016.

[4] Andreas Rauber, Tomasz Miksa, Rudolf Mayer, Stefan Proell. Repeatability and Re-Usability in Scientific Processes: Process Context, Data Identification and Verification. In *Proceedings of the 17th International Conference on Data Analytics and Management in Data Intensive Domains (DAMDID)*, Obninsk, Russia, October 2015.

[5] Tomasz Miksa, Ricardo Vieira, José Barateiro, Andreas Rauber. VPlan — Ontology for Collection of Process Verification Data, in *Proceedings of International Conference on Digital Preservation (iPres 2014)*, Melbourne, Australia 2014.

[6] Sandve G.K., Nekrutenko A., Taylor J., Hovig E. (2013). Ten Simple Rules for Reproducible Computational Research. *PLoS Comput Biol* 9(10): e1003285. https://doi.org/10.1371/journal.pcbi.1003285

[7] A. Rauber, A. Asmi, D. van Uytvanck, S. Proell. Data Citation of Evolving Data — Recommendations of the RDA Working Group on Data Citation. Oct. 20, 2015, RDA.

[8] A. Rauber, Ari Asmi, Dieter van Uytvanck, Stefan Proell. Identification of Reproducible Subsets for Data Citation, Sharing and Re-Use. *Bulletin of IEEE Technical Committee on Digital Libraries* (TCDL), Vol. 12, 2016.

[9] Stefan Proell, Kristof Meixner, Andreas Rauber. Precise Data Identification Services for Long Tail Research Data. In *13th International Conference on Digital Preservation (iPRES 2016)*, 2016. https://doi.org/10.6084/m9.figshare.3847632.v1

[10] Borgman, Christine L. *Big Data, Little Data, No Data: Scholarship in the Networked World — Who is in Charge of Data Quality?* MIT Press, 2015

[11] Ram, Karthik. Git can facilitate greater reproducibility and increased transparency in science. *Source code for biology and medicine 8.1* (2013): 1.

About the Authors

Stefan Pröll is a researcher at SBA Research focusing on Digital Preservation. He received a master's degree in Databases and Information Systems from the University of Innsbruck in 2009 and a master's degree in Business Engineering from the Vienna University of Technology in 2016. Currently he is working on his Ph.D. thesis in the area of Computer Science specializing on data citation, also at the Vienna University of Technology. Stefan is a co-chair of the RDA Working Group on Data Citation and an active member of the RDA Working Group on Data Security and Trust.

Andreas Rauber is Associate Professor at the Department of Software Technology and Interactive Systems (ifs) at the Vienna University of Technology (TU-Wien). He furthermore is president of AARIT, the Austrian Association for Research in IT, a Key Researcher at Secure Business Austria (SBA-Research) and Co-Chair of the RDA Working Group on Data Citation. He received his MSc and PhD in Computer Science from the Vienna University of Technology in 1997 and 2000, respectively. In 2001 he joined the National Research Council of Italy (CNR) in Pisa, followed by an ERCIM Research position at the French National Institute for Research in Computer Science and Control (INRIA), at Rocquencourt, France, in 2002. From 2004-2008 he was also head of the iSpaces research group at the eCommerce Competence Center (ec3). His research interests cover a broad scope of data science ranging from data analysis to data stewardship, with a specific focus on text and music information retrieval, as well as machine learning and digital preservation.