Health Informatics Meets eHealth
G. Schreier et al. (Eds.)

doi:10.3233/978-1-61499-645-3-245

Secondary Use of Claims Data from the Austrian Health Insurance System with i2b2: A Pilot Study

Florian ENDEL [a,1] , Georg DUFTSCHMID [b]
[a] *Vienna University of Technology*
[b] *Medical University of Vienna, Center for Medical Statistics, Informatics, and Intelligent Systems*

Abstract. Background: In conformity with increasing international efforts to reuse routine health data for scientific purposes, the Main Association of Austrian Social Security Organisations provides pseudonymized claims data of the Austrian health care system for clinical research. Objectives: We aimed to examine, whether an integration of the corresponding database into i2b2 would be possible and provide benefits. Methods: We applied docker-based software containers and data transformations to set up the system. To assess the benefits of i2b2 we plan to re-enact the task of cohort formation of an earlier research project. Results: The claims database was successfully integrated into i2b2. The docker-based installation approach will be published as git repository. The assessment of i2b2's benefits is currently work in progress and will be presented at the conference. Conclusions: Docker enables a flexible, reproducible, and resource-efficient installation of i2b2 within the restricted environment implied by our highly secured target system. First preliminary tests indicated several potential benefits of i2b2 compared to the methods applied during the earlier research project.

Keywords. Health Information Systems, Medical Records Systems

1. Introduction

Reuse of routinely collected administrative claims data for clinical research gains momentum all over the world (e.g. [1-3]). This is also the case in Austria, where the Main Association of Austrian Social Security Organisations [2] (HVB) provides a collection of claims data called GAP-DRG[3] for research purposes. Based on this database numerous research projects have been conducted (e.g. [4-7]), which typically relied on manually developed database queries and individually applied statistical procedures.

The goal of the present work was to examine whether an integration of the GAP-DRG database into the i2b2 clinical research framework [8-10] would be possible. This should provide a consistent and more efficient working environment for the before-mentioned kinds of projects. Due to the size (amongst other data, there are nearly $5 * 10^8$ ambulatory services and about $1{,}9 * 10^8$ medication prescriptions and dispensings of about $11 * 10^6$ patients documented for two years) and the complex structure (59

[1] Corresponding Author: Florian Endel, Vienna University of Technology, Institute for Analysis and Scientific Computing, Wiedner Hauptstraße 8-10, 1040 Wien. E-Mail: florian.endel@tuwien.ac.at

[2] in German "Hauptverband der österreichischen Sozialversicherungsträger"

[3] GAP-DRG is an abbreviation of one of the first larger research projects based on the data collection called "General Approach for Patient-oriented Ambulant DRGs"

normalized tables with cleanly defined relational constraints for two years' worth of data) of the GAP-DRG database, we expected the process to be challenging. Also, GAP-DRG is embedded in a highly secured hardware infrastructure with several restrictions that would likely complicate our work.

In the present work we report on the experiences we have made so far with integrating the GAP-DRG database into i2b2.

1.1. The GAP-DRG database

The GAP-DRG database comprises pseudonymized claims data from the Austrian healthcare system covering about 97% of the Austrian population[4]. Pseudonymized data about medication prescriptions and dispensings, sick leaves, ambulatory visits and inpatient episodes are linked and accompanied by demographics as well as various metadata (e.g. spatial data and diagnostic schemas). Additionally, derivated information resulting from previous projects, like for example diagnoses determined from prescriptions data [11], are integrated tightly.

The database was set up with data from all 19 Austrian social insurance institutions covering the years 2006 and 2007. Much knowledge about the data generating process, i.e. the Austrian healthcare system and it's reporting procedures was gathered during this first era of GAP-DRG. Among other challenges, methods for data quality assessment [12] and record linkage [13,14] had to be explored and developed. Additionally, most elements of the infrastructure supporting the database, including secured severs with remote desktop environments for researchers, a virtual private network with two-factor authentication and a wiki system for documentation, were created. In the course of a second delivery, data from Lower Austria's insurance carrier from the years 2008 to 2011 were added. Data quality is controlled routinely and project-wise [12,15].

Overall, the research database GAP-DRG covers the majority of all healthcare services of nearly the entire Austrian population for the years 2006 and 2007. However, some shortcomings (e.g., lacking diagnoses from the outpatient sector) and blind spots (e.g., lacking data from ambulatory care in hospitals) are known and have to be minded in research projects.

Since GAP-DRG constitutes a system that has evolved over time, incorporating various data sources, the resulting data model became rather complex, hard to handle and update. Furthermore, admission control on a user level to improve data governance is hardly possible in the current system.

1.2. The i2b2 framework

Informatics for Integrating Biology and the Bedside (i2b2) [8] is the name of the NIH[5] funded National Center for Biomedical Computing (NCBC) in Boston, Massachusetts and one of their products, a clinical data informatics framework[6] [16]. The source code of the i2b2 platform was released in 2007 under their own open software license.

i2b2 was originally developed for reusing electronic health records (EHR) and other patient centered data for research by providing integration and secure presentation of

[4] About 3% of the population are covered by other insurance carriers (e.g., municipalities, religious orders, unemployment service) and are not included in GAP-DRG.

[5] National Institute of Health: nih.gov

[6] In the context of this paper, the term i2b2 refers to the software framework only.

these data. It is designed to be easily extendable by splitting it up into several modules called cells, which communicate via a web service based on SOAP and REST calls [9]. The conglomeration of these loosely coupled cells constitutes the i2b2 Hive [17]. Data and configuration information are stored in a database system. Additionally, web and desktop clients as well as sample data and VMware images with pre-installed software are available.

Even for basic functionality, several cells must be deployed. Most important for this paper, the CRC ("Clinical Research Chart") cell is the central data repository and the ONT ("Ontology") cell manages the specific hierarchical metadata structure. Additionally, there are cells for e.g. project management, user authentication and system configuration. The central part of the CRC cell holding the core data is modelled as an entity, attribute, value (EAV) [18] schema. Several dimension tables containing information about patients, providers, visits, concepts and modifiers are related with the central fact table. The ONT cell requires a specific hierarchical metadata structure describing the fact and dimension tables as well as providing the foundation for the graphical representation in client software. Besides these basic cells, several specialized cells, e.g. for handling genomic data and natural language processing (NLP), are available.

2. Methods

Our first step was the implementation of the i2b2 platform under the specific circumstances and requirements of the GAP-DRG infrastructure. The second step was to transform the GAP-DRG data into the format prescribed by i2b2 and to load it into the platform. In the third step, which we are currently working on, we aim to examine the benefits of i2b2 in comparison to applying manually developed database queries for typical tasks of clinical research projects. For this purpose, we plan to re-enact an earlier project [19] that was conducted without i2b2.

2.1. Installation of i2b2

The installation process of the i2b2 hive is known to be complex [16] and for our desired combination of operating system (Linux) and database (PostgreSQL) the official installation guide is affected by several errors. The installation is further complicated by the fact that software components of rather specific and partly long outdated releases are required to work together seamlessly.

Furthermore, the infrastructure surrounding the GAP-DRG database, where we aim to deploy the i2b2 platform, consists of highly secured and therefore conservatively managed Linux (CentOS 5) servers without internet access and administration rights for the authors. According to system administration policy we were not allowed to install i2b2 in a traditionally virtualized environment or even use the readily available image.

In recent years operating system level virtualization, also known as software containers, like docker[7] [20,21] evolved rapidly. A docker container is an instance of a docker image running (ideally) exactly one single application. Images can be defined and built using Dockerfiles, which consist mostly of Linux shell scripts installing and configuring the requested application. Docker images require the docker environment

[7] docker.com

only and are largely independent of the underlying Linux system. Therefore, the deployment of containers consists primarily of exporting the prepared images and importing them on the targeted server. Additionally, some minor adjustments of the docker-compose configuration (a tool for container orchestration) have to be implemented, considering aspects of the hardware resources and network configuration.

Main advantages of a containerized solution are the reproducible and automatic building process, the low resource overhead of a running container and integrated options for serialization and transportation of images. After considering all these aspects, we arrived at the decision that software containers based on docker would be a favourable solution for bringing i2b2 to the GAP-DRG infrastructure.

2.2. Extraction, transformation, and loading GAP-DRG data into i2b2

Data from the GAP-DRG database was extracted, transformed and loaded (ETL process) for development and testing. As the development environment did not fulfil the security requirements for handling the original GAP-DRG data, data exported for testing were randomly distorted (gender, year of birth, dates of service utilization, association between individuals and claimed services). This allowed us to test with insensitive data that still resembled the original structure and size. Except for the random distortion, the developed ETL procedures were designed and tested as reproducible processes. This subsequently allowed us to re-enact the ETL process with the original data on the target system.

2.3. Examination of i2b2's benefits in a real-world GAP-DRG research project

In an earlier GAP-DRG based research project we analyzed by means of manually developed database queries and statistical methods whether young parenthood may be seen as a risk factor for myocardial infarctions [19]. We now plan to re-enact the task cohort formation and exploration for this project within the new i2b2 framework. This should help us to assess the benefits of i2b2 in this context by means of a "before-after comparison".

At the moment, manually developed SQL scripts are the most commonly used way to form and explore cohorts in GAP-DRG. We aim to examine how users experience working with i2b2, to what extent the currently applied methods that require skilled and experienced programmers can be facilitated or even replaced by the new graphical user interface provided by i2b2, and whether the group of GAP-DRG users may be broadened by means of i2b2. As the current user group is limited, the assessment of i2b2's benefits will be conducted with a small sample of persons, namely (i) two medical doctors being experts on the Austrian healthcare system without programming experience but knowledge of GAP-DRG, (ii) two experienced database developers with knowledge of GAP-DRG, and (iii) two persons with computer skills but without prior knowledge of GAP-DRG will take part in a black-box testing approach. After a short introduction to the system, they will try to execute basic cohort formation and exploration tasks with i2b2.

Several aspects concerning accessibility, ease of use, required knowledge, total amount of time taken to acquire a result, ratio of productive work and undirected searching, and number of deadlocks where help is required will be extracted from a screen recording and a short interview after the tasks have been accomplished. Additionally, the database developers will be asked to implement the same tasks with current tools to gather hard facts and personal feedback on actual differences between both approaches.

Summarizing, we plan to gather information on the differences between current workflows and i2b2 by involving three types of possible users. Advantages and shortcomings of the new system should allow us to find a suitable niche for the new tool, give information on an integrated workflow and might even help to improve both established and novel approaches.

3. Results

3.1. Installation of i2b2

i2b2 was successfully installed and tested data using software containers based on docker. As typical docker containers only run a single application, we separately prepared three main components of the i2b2 framework. We accompanied the Dockerfiles with several supporting shell scripts and developed and tested them to handle all i2b2 1.7.04 to 1.7.06 releases. These scripts allow various combinations of software components and versions to be chosen during the building process, preparing for prompt and flexible reactions to new software releases.

First, we deployed i2b2 to the PostgreSQL database of GAP-DRG. We created appropriate PostgreSQL instances for development and testing. To cope with the expected amount of data, a columnar storage [22] extension (cstore_fdw from citusdata) that was implemented as foreign data wrapper, was integrated with the official docker images of PostgreSQL 9.3 and PostgreSQL 9.4 as well as in the database of the GAP-DRG platform. Routines to load different versions of i2b2's test data and procedures were added as well.

Second, i2b2's web client and administration interface was installed on an official docker image of Apache httpd and PHP 5.6. Several extensions (i.e. ExportXLS, GIRI, CARE Concept Demographic Histograms, ARE Concept Observation Tally Histograms, WISE Annotator, WISE Searcher, IDRT Web Client Plugin [23]) were integrated and configured during the build process.[8]

Third, the i2b2 hive was prepared using a vanilla CentOS 7 docker image. In addition, Ubuntu 14.10 and baseimage-docker from Phusion were equipped and tested to run the readily installed hive as well. An i2b2 image including the very specific but largely outdated software products from the installation manual was implemented as intended by the developers. Additionally, docker images providing an i2b2 installation based on all later main releases of Java, JBoss Application Server known as Wildfly, Apache Axis and Apache Ant in various combinations were prepared and tested successfully. All i2b2 core cells, the patients count plugin for the CRC cell as well as the GIRI (Generic Integration of R into i2b2) [24] and IDRT [23] extension were integrated and configured during the build process.

Especially the integration of the R extension GIRI required special attention. GIRI was deployed as an i2b2 cell inside the hive. It exchanges data and plots with the webclient extension using the local file system by writing directly to the webserver's folders. As the hive and the webserver were split into different docker images in our installation, we had to employ sharing volumes between both containers.

[8] The IDRT plugin could not be tested successfully even after receiving help from the original authors.

In addition to these three core images, several supporting images were created to ease the database setup, configuration of several runtime parameters, create and manage new projects as well as create backups. Also, monitoring and logging software as well as database clients were prepared as docker images. docker-compose was applied to calibrate and coordinate all these independent components, configure shared volumes, connect network interfaces and allocate hardware resources.

We made some inconvenient experiences with existing public i2b2 resources. Besides unclear and erroneous sections of the installation guide and hard to interpret error messages, we had to tackle and patch actual software bugs. By providing means to flexibly combine various software and database versions as well as test patches in a reproducible way, our docker-based building routines helped to find and correct the flaws.

A task that we are still working on is the installation of i2b2's desktop client. Whereas the existing Windows 7 version works fine, no official build exists for Ubuntu Linux that we run our test system on. A self-prepared build proved unstable. On the GAP-DRG CentOS 5 Linux system we have not yet been able to build an executable desktop client due to lacking administrator privileges. This is planned for the next system update cycle by the GAP-DRG administrator. As a workaround we used plain database queries for the ETL process and the i2b2 webclient as main user interface.

3.2. Extraction, transformation, and loading of GAP-DRG data into the i2b2 platform

All data that were used in [19] were exported from GAP-DRG. The extract was then transformed to the CRC cell's entity-attribute-value (EAV) data model [18]. As required by i2b2, contacts (in our case claimed healthcare services such as hospital episodes) were modelled as entities, whereas their properties (e.g. time interval, diagnoses, etc.) were represented as attributes and corresponding type-specific values. Also modifiers, which allow to include additional information about an attribute (e.g., whether a diagnosis is regarded as main or additional diagnosis) were applied. Further, metadata from the GAP-DRG database (e.g., diagnosis and medication-related terminologies, demographics, and details of the financing system) were transformed into the defined schema.

Next, the ontology and the hive's configuration were transferred to the targeted database inside GAP-DRG. After completing the ETL test phase with the distorted data in our test environment, all project related data (stored in tables of the CRC cell) were extracted, transformed, and loaded without any distortion inside the secure target environment. Finally, a working i2b2 instance including several extensions and a project based on data from the GAP-DRG research database was prepared for deployment. After successful tests, the deployment was carried out on a separate development server without problems.

3.3. Examination of i2b2's benefits in a real-world GAP-DRG research project

Currently we are still working on re-enacting the tasks of cohort formation and data export for statistical analysis for the project described by Endel et al. [19] within the new i2b2 framework. First preliminary tests of the task cohort formation indicated several useful features of i2b2 such as iterative and interactive discovery of research cohorts, collaboration between distant individuals by enforcing role based access limitation, security policies and automatic documentation at the same time. We aim to present a detailed comparison of the handling of the two tasks with either conventional manually developed database queries or the i2b2 framework at the conference.

4. Discussion

The goal of this work was to examine, whether a large database containing claims data from the Austrian healthcare system could be integrated into the i2b2 framework and to clarify resulting benefits compared to previous methods of data handling. Despite extensive available i2b2 background resources, the installation process proved to be complex. Besides having to deal with erroneous documentation and outdated official i2b2 system components, we had to deviate from standard installation procedures due to restrictions implied by our highly secured target hardware infrastructure. Under the given circumstances, we decided to organize the different i2b2 components in separate docker-based software containers. Hereby, a flexible, reproducible, and resource-efficient installation of i2b2 was achieved. To ease the installation process for other novel i2b2 adopters, we plan to make the complete source code of our docker-based approach available as git repository on github.com/FlorianEndel.

The transformation of the original data model into i2b2's EAV model was complicated due to unclear documentation but overall rather straight forward. The size of the GAP-DRG database and the resulting vast amount of data accumulated in the fact table required the application of novel technical approaches utilizing PostgreSQL's foreign data wrappers and a column oriented storage engine. The creation of the metadata tree for the ontology cell and the integration of other annotating information proved to be cumbersome. More documentation on the semantics and interaction between tables and cells would have been of great help. After gathering first experiences, extending and optimizing the integrated data was a much swifter process.

The comparison of data handling with i2b2 compared to previous methods is currently work in progress. We aim to report corresponding strengths and limitations of i2b2 at the conference.

References

[1] H. H. Pham, D. Schrag, A. S. O'Malley, B. Wu, and P. B. Bach, "Care patterns in Medicare and their implications for pay for performance," *New England Journal of Medicine*, vol. 356, no. 11, pp. 1130–1139, 2007.

[2] S.-H. Cheng, Y.-F. Hou, and C.-C. Chen, "Does continuity of care matter in a health care system that lacks referral arrangements?," *Health Policy and Planning*, vol. 26, pp. 157–162, Mar. 2011.

[3] F. Chini, P. Pezzotti, L. Orzella, P. Borgia, and G. Guasticchi, "Can we use the pharmacy data to estimate the prevalence of chronic conditions? a comparison of multiple data sources," *BMC Public Health*, vol. 11, p. 688, Sept. 2011.

[4] G. Duftschmid, W. Dorda, G. Endel, K. Fröschl, W. Gall, W. Grossmann, and M. Hronsky, "Fragmentation of diabetes treatment in Austria - an indicator for the need for shared electronic health record systems," *Studies in Health Technology and Informatics*, vol. 180, pp. 667–671, 2012.

[5] S. Thurner, P. Klimek, M. Szell, G. Duftschmid, G. Endel, A. Kautzky-Willer, and D. C. Kasper, "Quantification of excess risk for diabetes for those born in times of hunger, in an entire population of a nation, across a century," *Proceedings of the National Academy of Sciences*, Mar. 2013.

[6] S. K. Sauter, L. M. Neuhofer, G. Endel, P. Klimek, and G. Duftschmid, "Analyzing healthcare provider centric networks through secondary use of health claims data," in *Biomedical and Health Informatics (BHI), 2014 IEEE-EMBS International Conference on*, pp. 522–525, IEEE, 2014.

[7] C. Rinner, S. K. Sauter, L. M. Neuhofer, D. Edlinger, W. Grossmann, M. Wolzt, G. Endel, and W. Gall, "Estimation of severe drug-drug interaction warnings by medical specialist groups for Austrian nationwide eMedication:," *Applied Clinical Informatics*, vol. 5, no. 3, pp. 603–611, 2014.

[8] S. N. Murphy, M. E. Mendis, D. A. Berkowitz, I. Kohane, and H. C. Chueh, "Integration of clinical and genetic data in the i2b2 architecture," in *AMIA Annual Symposium Proceedings*, vol. 2006, p. 1040, American Medical Informatics Association, 2006.

[9] S. N. Murphy, M. Mendis, K. Hackett, R. Kuttan, W. Pan, L. C. Phillips, V. Gainer, D. Berkowicz, J. P. Glaser, I. Kohane, and others, "Architecture of the open-source clinical research chart from Informatics for Integrating Biology and the Bedside," in *AMIA annual symposium proceedings*, vol. 2007, p. 548, American Medical Informatics Association, 2007.

[10] V. G. Deshmukh, S. M. Meystre, and J. A. Mitchell, "Evaluating the informatics for integrating biology and the bedside system for clinical research," *BMC Medical Research Methodology*, vol. 9, no. 1, p. 70, 2009.

[11] P. Filzmoser, A. Eisl, and F. Endel, ATC –> ICD: Determination of the reliability for predicting the ICD code from the ATC code. 2009.

[12] F. Endel, "Understanding data quality in linked administrative data," in *International Health Data Conference 2014*, (Vancouver), 2014.

[13] F. Endel, G. Endel, and N. Pfeffer, "PRM34 Routine Data in HTA: Record Linkage in Austrias GAPDRG Database," *Value in Health*, vol. 15, p. A466, Nov. 2012.

[14] F. Endel, G. Endel, B. Weibold, and H. Katschnig, "Health service record linkage in a situation of multiple social health insurance institutions: the case of Austria," in *Proceedings of The methodological challenges of record linkage*, (University of St. Andrews, Scotland), Sept. 2011.

[15] F. Endel and H. Piringer, "Data Wrangling: Making Data Useful Again," in *MATHMOD 2015 Vienna -Abstract Volume*, vol. 8, (Vienna University of Technology, Vienna, Austria), pp. 111-112, ARGESIM and ASIM, German Simulation Society, Div. of GI - German Society for Informatics / Informatics andLife Sciences, 2015.

[16] E. K. Johnson, S. Broder-Fingert, P. Tanpowpong, J. Bickel, J. R. Lightdale, and C. P. Nelson, "Use of the i2b2 research query tool to conduct a matched case–control clinical research study: advantages, disadvantages and methodological considerations," *BMC medical research methodology*, vol. 14, no. 1, p. 16, 2014.

[17] S. N. Murphy, G. Weber, M. Mendis, V. Gainer, H. C. Chueh, S. Churchill, and I. Kohane, "Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2)," *Journal of the American Medical Informatics Association*, vol. 17, pp. 124–130, Mar. 2010.

[18] P. M. Nadkarni, "QAV: querying entity-attribute-value metadata in a biomedical database," *Computer Methods and Programs in Biomedicine*, vol. 53, pp. 93–103, June 1997.

[19] F. Endel, S. Sauter, L. Koller, A. Niessner, and G. Duftschmid, "Reusing claims data to assess parenthood as risk factor for myocardial infarction," *Studies in Health Technology and Informatics*, vol. 210, pp. 979–979, 2015.

[20] J. Fink, D. S. Librarian, and M. University, "Docker: a Software as a Service, Operating System-Level Virtualization Framework," *The Code4Lib Journal*, July 2014.

[21] D. Merkel, "Docker: Lightweight Linux Containers for Consistent Development and Deployment," *Linux J.*, vol. 2014, Mar. 2014.

[22] D. J. Abadi, P. A. Boncz, and S. Harizopoulos, "Column-oriented database systems," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1664–1665, 2009.

[23] C. R. K. D. Bauer, T. Ganslandt, B. Baum, J. Christoph, I. Engel, M. Löbe, S. Mate, S. Stäubert, J. Drepper, H.-U. Prokosch, A. Winter, and U. Sax, "Integrated Data Repository Toolkit (IDRT): A Suite of Programs to Facilitate Health Analytics on Heterogeneous Medical Data," *Methods of Information in Medicine*, vol. 54, Nov. 2015.

[24] Weinlich, B., Mate, S., Prokosch, H.U., Ganslandt, T., and Toddenroth, D., "„R-Scriptlets" für i2b2-Endanwender," 2014