German Medical Data Sciences: Shaping Change – Creative Solutions for Innovative Medicine
R. Röhrig et al. (Eds.)

doi:10.3233/SHTI190807

Verifying Data Integration Configurations for Semantical Correctness and Completeness

Mark R. STÖHR[a,1], Andreas GÜNTHER[a] and Raphael W. MAJEED[a]
[a] *UGMLC, German Center for Lung Research (DZL), Justus-Liebig-University, Giessen, Germany*

Abstract. Data integration is the problem of combining data residing at different sources and providing the user with a unified view of these data. In medical informatics, such a unified view enables retrospective analyses based on more facts and prospective recruitment of more patients than any single data collection by itself. The technical part of data integration is based on rules interpreted by software. These rules define how to perform the translation of source database schemata into the target database schema. Translation rules are formulated by data managers who usually do not have the knowledge about meaning and acquisition methods of the data they handle. The professionals (data providers) collecting the source data who have the respective knowledge again usually have no sufficient technical background. Since data providers are neither able to formulate the transformation rules themselves nor able to validate them, the whole process is fault-prone. Additionally, in continuous development and maintenance of (meta-) data repositories, data structures underlie changes, which may lead to outdated transformation rules. We did not find any technical solution, which enables data providers to formulate transformation rules themselves or which provides an understandable reflection of given rules. Our approach is to enable data providers understand the rules regarding their own data by presenting rules and available context visually. Context information is fetched from a metadata repository. In this paper, we propose a software tool that builds on existing data integration infrastructures. The tool provides a visually supported validation routine for data integration rules. In a first step towards its evaluation, we implement the tool into the DZL data integration process and verify the correct presentation of transformation rules.

Keywords. Quality improvement, metadata, automatic data processing, data accuracy, data aggregation, communication barriers, data visualization

1. Introduction

1.1. Background

Data integration is the problem of combining data residing at different sources, and providing the user with a unified view of these data [1]. To harmonize multiple heterogeneous data sources is an important topic for research networks, registers and

[1] Corresponding Author, Mark R. Stöhr, UGMLC, Justus-Liebig-University, Klinikstraße 36, 35392 Gießen, Germany, E-Mail: mark.stoehr@innere.med.uni-giessen.de

other consortia planning data analyzes on large data pools. The complexity of data integration processes results not only from a technical point of view but also from the variety of professions whose contribution is required.

Future-oriented implementations of data integration processes have to meet various requirements in order to be able to interact with the global research community as well as to react to arising scientific questions. Wilkinson et al. formalized these requirements and introduced the FAIR principles (Findable, Accessible, Interoperable, Re-usable) as a guideline [2]. Expressiveness of data needs to be enriched with annotations, e.g. labels, standardized codes and hierarchical information, which guarantee unambiguousness and comparability. This can be achieved by establishing metadata repositories, which contain descriptive and relational information about the data repository [3–5].

It is not possible to set up a metadata catalogue once and expect it to meet all future requirements. Metadata is composed over time by domain experts and requirements to the matter addressed by data integration processes are growing with its use cases. Thus, an agile system development approach appears to be indispensable. For a single data collection, three technical components are involved in the data integration process: (1) the data source which is often historically grown and differs from other sources in structure, functionality and research scope. (2) The target data repository into which data from every source is to be integrated. (3) The metadata repository with descriptive information of the entries in the data repository.

While data source and metadata repository are not necessarily directly connected to each other, they are both connected to the data repository via an identifier. The technical operations to load source data into the data repository are executed by ETL (Extract, Transform, Load) software which extracts data from its source database, transforms it according to given rules and loads it into the target database.

To configure ETL software, knowledge of several different participants is needed. On the source side, there are *data owners*, *data providers* and *data managers*. Data providers collect and utilize data whereas data managers transform, store and export it. On the integration side, there are *coordinators* for the metadata repository development, *data managers* who transform and store the data and *software developers* providing and customizing the ETL software. While data managers come from a technical background, coordinators and users like physicians and medical documentalists have a medical background and the knowledge about the actual data acquiring processes.

Since every person involved is lacking knowledge from the other domains, mappings from source datasets to a harmonized dataset are fault-prone. Data providers are usually not able to compose formal transformation rules and data managers are often confronted with ambiguous data. After data managers composed the mapping rules, the respective data providers have to verify their correctness and completeness. This quality assurance is only possible, if the data provider understands the meaning of the technical realization. Besides semantic correctness, agile metadata development may cause changes affecting the ETL configuration. For example, a data element's temporary code may be replaced by a newly added code requested from a standardization organization. Therefore, data managers also have to check for newer versions of the metadata.

In this article, we address two issues impeding correct data integration: First, there is an indirect link between the data source and the metadata repository, which is established by the ETL configuration rules. This link is not transparent to data provider, since they have no access to the formal definitions. Thus, they cannot verify for

correctness and completeness. Second, data managers configuring the ETL mappings need a reflection of changes in the metadata that affect their configurations.

1.2. Requirements

To achieve solutions for the two identified issues, we need to provide transparency about the ETL process. We assume that an existing data integration infrastructure is already in place, including data sources, a metadata repository, a data warehouse and respective ETL processes. The same identifiers used in the data repository and to which source data is mapped are also available in the metadata repository. We also assume that changes to the metadata repository are available in processable form.

Every participant has to be able to retrieve the information they need in order to bring in their own expertise: Data providers must be able to verify the correctness and completeness of the composed mappings. Accordingly, they need to know how their data is mapped to the metadata catalogue. Thus, a tool is required to extract mapping information from the ETL configuration and visualize its binding to the metadata in an intuitively understandable way. All available context information from the metadata repository should be presented. This includes labels, descriptions, standardized codes and relations to other data elements like hierarchical information and "siblings". Additionally, the tool should indicate metadata changes, which affect the ETL configuration. The respective data managers must be able to understand the performed changes in order to update the configuration.

2. State of the art

We found several existing solutions targeting the integration of (clinical) data into data warehouses: Ong et al. [6] developed ETL software that uses ETL rule files that state the ETL mappings. They describe two different roles involved in the ETL process, so-called subject matter experts and database programmers. The subject matter experts are expected to "have extensive knowledge of the source and target schemas" and to be able to fill in rule entries. They admit, that for this process "certain SQL coding skill is needed" and "knowledge about the operators and functions of the DBMS is needed". The rules are entered in human-readable text-based files, since "GUI tools are not flexible enough and lack transparency". Pecoraro et al. [7] followed a similar approach by letting users enter rules in mapping files. They still require some technical background to perform this task. Additionally, their ETL software expects source data to be available in HL7 CDA and is only able to perform one-to-one and many-to-one mappings. Post et al. [8,9] developed a ETL software called Eureka!. They bypassed the need of varying mapping configurations by offering a stylesheet-based template in which source data has to be entered. The data-supplying user does not need to formulate relations of his data to the data repository. However, he is forced to put his data in a predetermined schema. With their setup, they realized "highly metadata-driven ETL processing". Vučković et al. [10] analyzed existing ETL modeling approaches and identified six basic transformation operations: Join, Equivalence, Equals, Concatenate, Condition and Constraint.

Additionally, there are generic tools for implementation and illustration of data integration processes, e.g. Talend Open Studio and Microsoft SQL Server Integration Services. They depict the transformation from source to target data schemas with various

operators like “merge” and “union”. However, they still lack sufficient clarity and context regarding the actual relations between data fields.

No presented solutions are applicable to scenarios in which physicians and medical staff are bound to their respective (clinical) input system and/or lack the required expertise to formalize mapping rules.

3. Concept

Our solution builds on top of existing infrastructure and complements the ETL configuration maintenance. Figure 1 is a schematic illustration of the intended infrastructure and workflow. In all ETL configurations, rules state how source fields are treated and how they will be represented in the target data schema. Those rules mostly follow the mindset of a database programmer and often contain cryptic terms and naming. Our approach is to move from the source data centered perspective to a target data centered one. That means, the required visualization tool displays the metadata repository’s elements with all their annotated information and for each of those elements, it illustrates their connection to the source data schema.

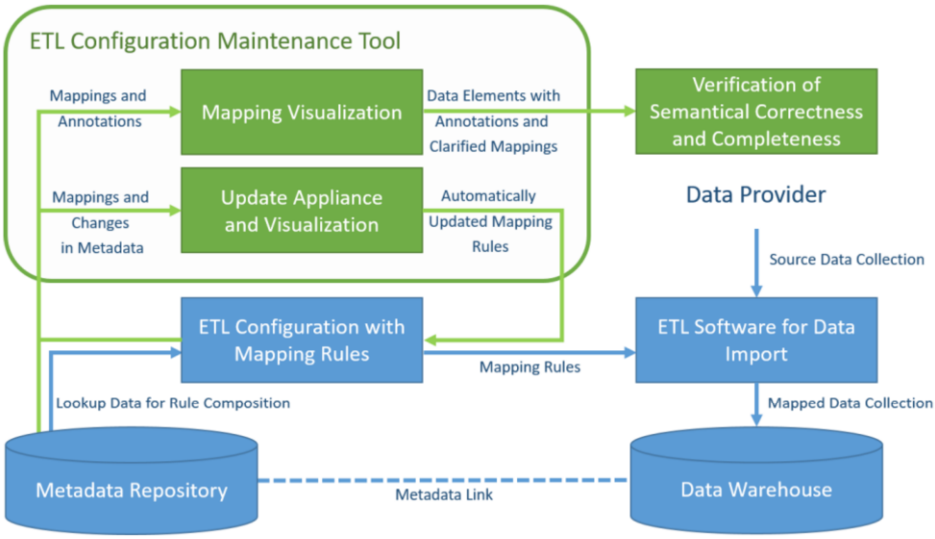

Figure 1. Intended infrastructure and workflow. Blue elements depict the ETL workflow for a single data collection. Green elements depict the additional verification process.

3.1. Available Information (Input)

There is no single solution for data integration infrastructures. In the introduction, we described three main components involved, the data source, the data repository and the metadata repository. Individually, they vary because of different purposes, histories, user capacities and preferences. To keep the concept as generic as possible, we assume that the metadata repository and the ETL configuration file provide the following abstract information in a processable way: The metadata repository stores annotations for each mapped source data field, e.g. labels, descriptions, standardized codes and

relations to other data elements. For each metadata element, we can access all changes that affect the ETL mappings, e.g. changes to the data element's datatype or even to its identifier. Furthermore, the composed ETL configurations include rules that can be categorized as shown in table 1 and table 2. These categories result from investigation (see state of the art section) as well as our own experiences based on over 20 integrated data sources.

Table 1. Categories of source to target data field transformations.

Field Transformation	Example
zero-to-one	Implicit information. Every patient in a cancer registry implies an existing cancer diagnosis, which is not explicitly mentioned.
one-to-one	Height of a person is a data element in both data schemas.
one-to-many	Diagnosis field filled with integers, each encoding the information for one diagnosis. A field containing information about both the biomaterial and its extraction type.
many-to-one	The data element "Operation and radiotherapy and chemotherapy" exists in the target data schema but each therapy is stored in a separate source field.
many-to-many	TNM classification in one field; the classification catalogue version in another field, which is required for a complete information. Both fields are mapped separately, but their semantic link remains.

Table 2. Categories of source to target data value transformations.

Value Transformation	Example
keep value as is	Height of a person is stored with unit centimeters in both data schemas.
calculate	Transform height in meters to height in centimeters by multiplication with "100".
lookup	If the source value is "1", then set value for concept "Pulmonary Hypertension" to true.
drop information	The concept/value is not part of the target data schema.

3.2. Presentation of Information (Output)

The data provider's view consists of metadata elements and mapping information. Metadata elements are shown with all available annotations. Mappings are all extracted from the ETL configuration file and presented as a combination of "Source Fields Names", "Source Value Condition", "Source Units" and "Target Value". Data managers also receive a list of changes made to the metadata, which affect their mappings. The tool offers them an automatically updated version of the ETL configuration.

4. Implementation

The German Center for Lung Research (DZL) uses i2b2 as its data warehouse software. Our ETL software expects source data to be available in delimited text format. The ETL configuration file is written in a proprietary XML format. The connection between metadata and the data repository is realized through concept codes, preferably standardized codes. For metadata composition, we use our Collaborative Metadata Repository (CoMetaR) tool. Metadata is expressed in RDF format and saved in a Git repository. Uploaded changes to the metadata are automatically propagated to the i2b2

ontology cell. Furthermore, metadata and its changes are available through a SPARQL interface. The CoMetaR browser (https://data.dzl.de/cometar/) visualizes the DZL ontology and lets users navigate and search through it.

We implemented the proposed ETL configuration maintenance tool in our CoMetaR browser. After uploading the configuration file through a file upload form, we show each mapping information under the respective metadata. Figure 2 shows an example where the field “Packyears” is mapped one-to-one to the DZL concept “Packyears” and non-empty values will be sent to the data warehouse without transformation. The field “Rauchverhalten” is mapped one-to-many to the DZL concepts “Active smoker”, “Ex-smoker” and others, depending on its string value. During the ETL process, the values are transformed through lookup from the respective string to “true”.

When uploading an ETL configuration file, all mapped concepts are automatically checked for outdated codes. If outdated, corresponding old and new codes are shown to the data manager and an updated version of the configuration file is offered for download.

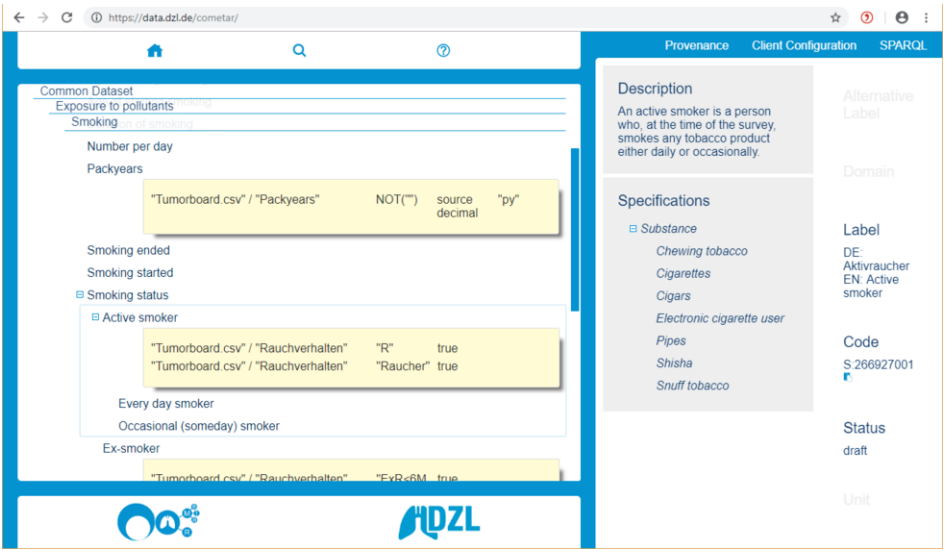

Figure 2. CoMetaR browser with loaded ETL configuration. To the left: hierarchically organized concept tree with ETL rules (yellow boxes). To the right: annotated metadata for the selected concept.

In order to achieve a comprehensive central data warehouse for the DZL, we already integrated more than 20 different data sources. This process includes the composition of mapping configuration files. In a first evaluation step for the proposed tool, we tested our implementation with the three largest of those configuration files containing 479, 256 and 158 mapping rules respectively. All rules are displayed correctly in the mapping visualization. However, our import software itself does not support many-to-one or many-to-many field transformations and no calculated value transformations.

5. Lessons learned (Discussion)

We introduced a new approach to analyze the quality of an ETL process regarding semantical correctness and completeness. Our claim is to enable data providers with less technical background to understand the formal definition of composed ETL mapping rules. Thereby, they could identify incorrect and/or missing mappings and inform the respective data managers. We also enable data managers to keep the ETL mappings up-to-date. We had to face some limitations when implementing our approach into our own infrastructure. Our ETL client can only use source files in delimited text format (e.g. CSV) and has limited support for many-to-one and many-to-many field transformations. Thus, for data integration we often need to pre-process data to meet our client's requirements. It is to check in how far mappings of pre-processed data are attributable to the original data source fields. In addition, an evaluation of the presented tool is needed in order to prove the concept's effectiveness.

6. Conclusion

We proposed the concept for a tool that supports data providers and data managers to keep ETL configurations for data integration processes correct, complete and up-to-date. Our approach is to display the metadata catalogue with all available annotations and highlight those, which are addressed by the ETL mapping rules. We provide transparency about how these complex rules operate on the source data. After verifying our implementation's ETL rule interpretation and presentation, we argue that additional context and rule clarification result in less ambiguity, more correct and complete mappings and thus higher data quality. We learned that preprocessing of source data might disguise the true origin of fields and values. Combined with the complexity of transformation rules, an intuitively understandable presentation of mappings can be challenging. Addressing this issue, we introduced a list of occurring mapping types and gave examples for a simplified visualization of their meaning regarding implicated source and metadata elements.

Conflict of Interest

The authors state that they have no conflict of interests.

References

[1] M. Lenzerini, Data integration: a theoretical perspective (2002), DOI: 10.1145/543613.543644.
[2] M.D. Wilkinson, et al., The FAIR Guiding Principles for scientific data management and stewardship, *Scientific data* **3** (2016), DOI: 10.1038/sdata.2016.18.
[3] H. Zhang, et al., An ontology-guided semantic data integration framework to support integrative data analysis of cancer survival, *BMC medical informatics and decision making* **18(2)** (2018), 41, DOI: 10.1186/s12911-018-0636-4.
[4] V. Stathias, et al., Sustainable data and metadata management at the BD2K-LINCS Data Coordination and Integration Center, *Scientific data* **5** (2018), DOI: 10.1038/sdata.2018.117.
[5] M. Dugas, et al., ODMedit: uniform semantic annotation for data integration in medicine based on a public metadata repository, *BMC medical research methodology* **16** (2016), 65, DOI: 10.1186/s12874-016-0164-9.

[6] T.C. Ong, et al., Dynamic-ETL: a hybrid approach for health data extraction, transformation and loading, *BMC medical informatics and decision making* **17(1)** (2017), 134, DOI: 10.1186/s12911-017-0532-3.
[7] F. Pecoraro, D. Luzi, F.L. Ricci, Designing ETL Tools to Feed a Data Warehouse Based on Electronic Healthcare Record Infrastructure, *Studies in health technology and informatics* **210** (2015), 929–933.
[8] A.R. Post, et al., Semantic ETL into i2b2 with Eureka!, *AMIA Joint Summits on Translational Science proceedings. AMIA Joint Summits on Translational Science* **2013** (2013), 203–207.
[9] A.R. Post, et al., Metadata-driven Clinical Data Loading into i2b2 for Clinical and Translational Science Institutes, *AMIA Summits on Translational Science Proceedings* **2016** (2016), 184–193.
[10] M. Vučković, et al., The Specification of ETL Transformation Operations based on Weaving Models, *International Journal of Computers Communications & Control* **7(5)** (2012), 968–975, DOI: 10.15837/ijccc.2012.5.1356.