EnviroInfo 2007 (Warschau)
Environmental Informatics and Systems Research
 ISBN: 978-3-8322-6397-3

Annotating and Integrating Legacy RDBMSs for the Utility Domain

Fabian Grüning[1], Mathias Uslar

Abstract

Since a few years the utility domain faces new challenges such as deregulation forced by the EU and its efforts towards improving efficiency of the energy markets and the increasing feed of renewable energy. These changes increase the amount of communication in and between companies, e.g. for trading at energy stock markets or for controlling offshore wind energy parks, thus increasing requirements for interoperability. Having a larger amount of small power plants like renewables and combined heat and power generation in the grid, the task for controlling them for assuring a stable grid gets much more complex. Therefore, many decisions have to be made automatically requiring a high data quality as decisions based on wrong data tend to be wrong. The proposed architecture provides a solution for both challenges.[2]

1. Motivation

The utility domain faces two different kinds of efficiency demands. On the one hand, the unbundling forced by the EU makes it possible for new participants to enter the energy markets by trading energy equated with the former companies of the energy domain. This action aims at economical efficiency. On the other hand, the ecological efficiency is furthered by agreements of many industrial countries (Kyoto protocol) mainly measured in reduction of CO2 emission. This goal can be achieved by increasing the efficiency of conventional power plants, establishing new power plants with higher efficiency rates like combined heat and power generation and making use of power plants based on renewable energy like wind and solar power plants. The result is an economically and ecologically more efficient energy domain through higher competition and more efficient energy conversion methods which are partially even CO2-neutral.

This scenario has a strong impact on the underlying ICT infrastructure. The unbundling and associated multitude of participants on the energy market requires a concept for communication between companies as well as the unbundled former energy companies. This can be established using shared knowledge in form of a domain ontology as bases for the communication like it is provided by the CIM (Uslar, 2006) but the task of mapping the companies legacy data structures still remains.

Assuring the stability of the electrical grid in the sense of stable frequency and voltage levels is a much more complex task if the grid is fed by many small power plants instead of a few major ones. It is impossible to manage this task manually; instead, approaches like virtual power plants are used to abstract from the complexity and handle many power plants as one virtual. This means that many control decisions have to be taken automatically, having a high demand in data quality as wrong input data in algorithms lead to wrong resource planning. This problem worsens if the simple infrastructure in terms of safety and liability the power plants are connected with each other (like ADSL connections) is taken into consideration so that the case of missing or outdated data is frequently faced.

We are going to introduce an architecture that addresses both challenges by providing a possibility of mapping an existing data management to an ontology. The data structures are, on the one hand, directly used for interoperability, but it is also possible to annotate data with regard to their data quality aspects as the used ontology is based on RDF (W3Cb, 2004).

[1] University of Oldenburg, e-mail: fabian.gruening@informatik.uni-oldenburg.de
[2] The project of this work has been established and funded by the German utility EWE AG.
Please visit http://www.ewe.de/.

The paper is structured as follows. First, we give an introduction to the focused aspects, mainly data quality and interoperability. Then we are going to take a deeper look at the requirements and meaning of those aspects and, thereby define the scenarios later used in the paper. Based on those scenarios we show the proposed architecture and explain its usefulness. Finally, we draw conclusions.

2. Introduction

Having already motivated the importance of the utility domain in the near future, regarding the impact on the environmental condition, especially the CO2 emissions, we are now going to focus on the aspects data quality and interoperability. These aspects are essential when it comes to fulfilling the required efficiency, both economically and ecologically as explained before.

2.1 Data Quality

Data quality plays an important part in keeping large, long-living information management systems, which can be found in utilities, healthy. As data quality management makes statements about data (metadata), a metadata management is needed additionally to the conventional information management to manage this aspect.

There are several indicators for high data quality which can be categorised and measured. A sound definition of those data quality dimensions and algorithms for measuring within the utility domain can be found in (Grüning, 2006), formulated as an ontology; the term "data quality mining" (DQM) is introduced. It covers data quality management through using algorithms of, amongst others, the data mining domain. Such algorithms need and produce, in proportion to the original, data large amounts of metadata, e.g. flags showing the degree to which a data tuple might be correct, or the value an original value is mapped to through the preprocessing phase of the DQM. There is a need of handling such metadata in a more flexible way, as long-living database management systems cannot simply be adjusted to such new requirements being not designed for from the beginning.

2.2 Interoperability in general

When we look at different systems and their life-cycles, we see that we cannot build a new all-in-one system due to both financial and technical reasons. The existing systems have a long lifespan and are currently supporting the day-to-day business. They cannot simply be omitted. But the aforementioned new requirements lead to new systems which are introduced into the overall system landscape. Often, they use new syntax and, even worse, different and new semantics for the existing concepts.

Because it is impossible and certainly not economically feasible or useful to get all enterprise's systems for running the utility from one vendor, the only possible solution is to integrate the new and existing systems. This problem can be addressed by interoperability. The overall idea is to establish common semantics when coupling the systems, using enterprise architecture integration (EAI) techniques. EAI is defined as a mixture of concepts, technologies and tools which in general combined lead to the integration of heterogeneous applications. One has to distinguish between EAI and middleware. Middleware is often used as a synonym for EAI but there is one important difference: EAI focuses on the integration of independent applications while middleware focuses on communication between application components.

In order to achieve interoperability, we have to establish common semantics and syntax. Our approach deals with the use of an RDF-based ontology for the domain. Using the ontology, we can semantically integrate our old relational databases and metadata and thus achieve better interoperability between old and new systems. The data model will be based on the Common Information model CIM as introduced later.

3. Scenarios

The two scenarios, handling metadata produced by data quality mining and interoperability by mapping legacy data schemas to ontologies, are introduced here so that details of those scenarios can be used in the further discussion.

3.1 DQM

In this section we are going to briefly introduce our data quality mining approach. We therefore introduce dimensions, algorithms and processes that define the approach and focus then on the metadata which is produced. We take up on this metadata in the next section where we show how this metadata can be handled by our architecture.

3.1.1 Dimensions, Algorithms, Processes

In the underlying DQM scenario, we cover four data quality dimensions (accuracy, completeness, consistency, and currency) with five different algorithms (preprocessing, logging, record linkage, classification, and outlier analysis) consecutively executed. Different processes are used for rating every single data quality dimension, the overall performance of the database's data quality, and, if necessary, provide suggestions for improving data quality.

The mentioned algorithms belong to both statistical analysis and data mining. The statistical algorithms are mainly used for preprocessing and evaluation of the results of data mining algorithms. Coming to the data mining algorithms, classification algorithms are utilized to build (generally speaking) models of the data in the data schemas by using input data which has been checked by domain experts and found correct. Using these metamodels, it is possible to identify wrong values by comparing the data stored in the data management system with the values calculated by the metamodel, even having a hint about the correct values through the classified one. Whenever those values differ from each another, a value is found suspicious to be a wrong value (Grüning, 2007).

This approach works best if there is a certain "semantical density" in the data schemas, i.e. having concepts in the ontological meaning without normalization, IDs, foreign keys etc. that only represent an abstraction from a class of entities of the real world.

Using preprocessing, it is also possible to identify duplicates. In this case, the classification task is to decide from the distances of elements of data tuples if those tuples represent a duplicate or not where a duplicate is defined as multiple representations of a real world entity. Again, a verified training set is needed to adjust the classification algorithm before duplicates can be detected.

3.1.2 Metadata

Every algorithm needs and leaves certain metadata that can be annotated to a certain representation of a real world entity, a certain instance or the instance's concept, speaking in ontological terms. A very short overview of the metadata ordered by DQM relevant algorithms is given in the following: The logging directly produces metadata by annotating the points of time when data gets modified. The preprocessing maps every data item to a real world value for further processing by the other algorithms, e.g. normalizing values. The record linkage builds groups of similar instances, rates the differences of single data items of those instances, and finally draws a conclusion which instances might represent the same real world entity as described before. The classification builds a metamodel to every concept, expressing its characteristics and discovering a concept's instances that diverge from those characteristics, therefore detecting wrong

data items, again as described above. Both the metamodels and the outliers are annotated to the concepts and instances, respectively. The outlier analysis, mainly used for analysing the outcomes of the other algorithms, simply detects outliers by using a clustering approach.

3.2 Common Information Model (CIM)

Coming back to the interoperability aspect, our approach is very familiar to the common EAI approaches integrating systems using the databases and the integration layer. Unlike traditional approaches, we do not integrate on the scheme layer itself but use modern techniques like RDF and OWL and ontologies in order to integrate the data semantically and provide cleansing in one step.

The Electric Power Research Institute EPRI from the USA has established the so called CIM, which has become an international standard by the International Electrotechnical Commission (IEC). The CIM can be seen as a domain ontology for the electric sector, standardizing objects, attributes and relations between objects. Most of the important objects for the electric utility are contained within the CIM. Furthermore, different containments like electrical, logical and physical containment exist. The model can be serialized in different ways. For message-based coupling of information systems, it is possible to use an XML serialization to create payloads based on CIM-semantics. We can also use RDF-based serialization in XML to provide proper rich semantics to serialize the representation of an electric power grid. The latest serialization is an OWL (Web Ontology Language) model of the CIM which can be used in different scopes. Our usage of the model will be explained later.

In summary, the CIM provides both semantics and proper serializations to use the data integration paradigm based on meta annotation and RDF semantics, Furthermore, richer semantics for the overall CIM can be achieved using the OWL model.

4. Architecture for Annotation and Integration

We now introduce an architecture that allows handling for both explained scenarios, the annotation of metadata produced by data quality mining and achieving interoperability through mapping from legacy database management schemas to standardized semantics. As the architecture heavily depends on RDF (W3Cb 2004) as technique for formulating ontologies, we will first give an introduction to RDF.

4.1 Introduction: RDF

Unlike the data schema used in relational database management systems, RDF relies on statements consisting of subjects, predicates, and objects. Therefore, an RDF-triple (S, P, O) is a statement and RDF data management is a set of those statements. RDF Schema (W3Ca 2004) defines classes and properties, where, for each property, valid domains and ranges have to be defined. Instances are implementations of classes with values for properties, whereas domains can also be simple data types like integers or strings. Classes and instances can be subjects and objects in the statements, properties are always predicates. All elements have namespaces so that naming collisions are prevented. Furthermore, every element can be identified, even beyond the local data management system. Through these mechanisms, it is possible to make statements about remote instances as their names and IDs can be used as subjects in RDF statements with every RDF data management system which means that annotations can be made.

4.2 Proposed Architecture

We introduce a method for both handling metadata and improving interoperability by using mapping techniques with the destination target RDF. In this scenario, there are several requirements imposed by

the metadata management and the improvement of interoperability: the utilities' legacy databases have to be embedded into the new architecture, a mapping has to be compiled that maps the data of the relational database to an either generic RDF equivalent (i.e. tables to concepts and attributes to properties) or to a domain ontology like the CIM (Uslar 2006), and, finally, a native RDF repository has to be provided for managing the metadata itself (Figure 4.2.1).

These requirements are met by the use of the D2R Server (Bizer, 2004) as a server that integrates the data of legacy relational database management systems by mapping from data schema to RDF as defined in a mapping using D2RQ (Bizer 2004). Again, this mapping can either be generic or domain specific using e.g. CIM. In the latter case, there is a synergy as such a mapping provides interoperability advantages regarding B2B and A2A.

The server can be queried, among others, by SPARQL (Prud'hommeaux, 2006). It provides a uniform way to make legacy database systems available to semantic web applications that use shared knowledge modelled as an ontology as the foundation of their communications and, therefore, providing improved interoperability. The mapping provides an "RDF view" to the data of the legacy databases. This view can also be used to use the CIM as originally intended by the IEC and to facilitate the exchange of power grid topologies using the native RDF format.

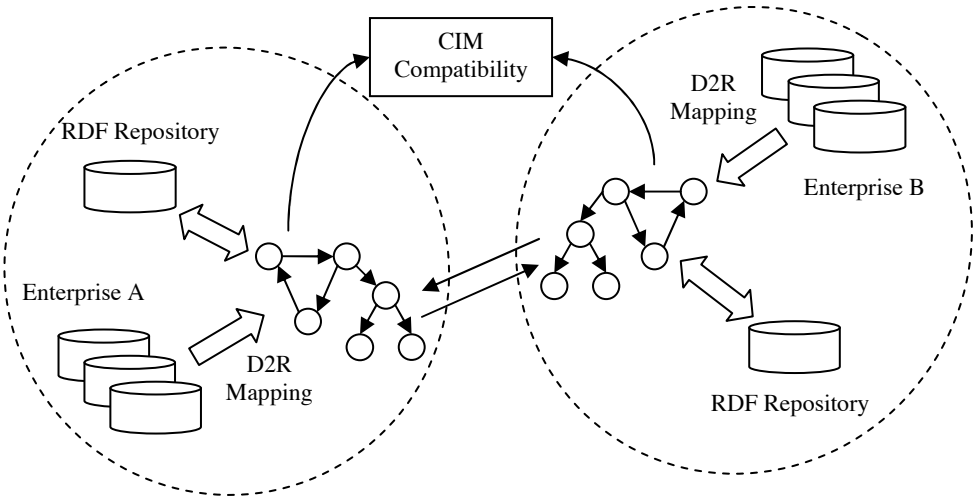

Figure 4.2.1: Proposed architecture for mapping legacy database systems to ontologies

Using this mechanism, every RDF instance obtains a URI which makes it uniquely identifiable, so they can directly be annotated using RDF statements. These annotations are stored in the native RDF repository Sesame 2.0 (Kampman, 2007)

4.3 D2R in Detail

The following figures show an example mapping from a relational database schema to the CIM (example taken from Martens 2007). The example models use the reservoir of a hydroelectric power plant. We first show the original database schema (Figure 4.3.1), then the ontology the schema gets mapped to (Figure

4.3.2) and finally the mapping which is described as an ontology itself (Figure 4.3.3) written in N3. Beware that the namespace "vocab" is set to the CIM namespace in the prologue of the mapping.

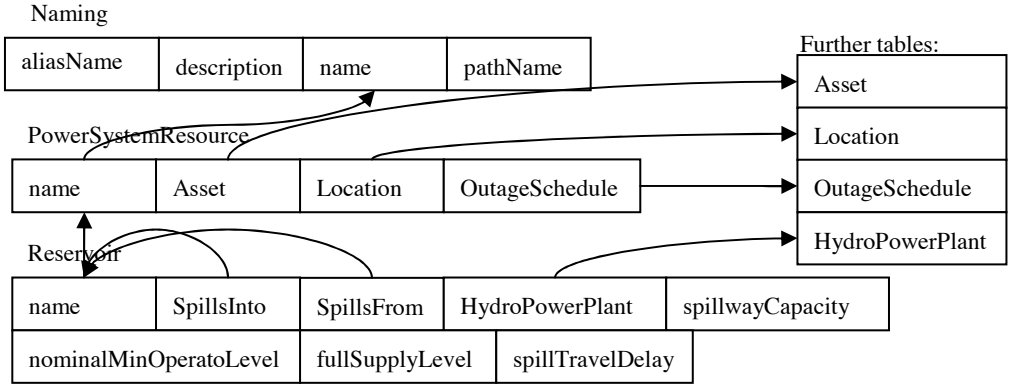

Figure 4.3.1: Relational database schema of legacy database management system

Taking a closer look at the mapping, a concept gets mapped via a ClassMap instance that defines the source, the distinct name and the concept to be mapped itself. The second statement defines the label instances are associated with. This is not the technical description of the instance as explained before, but rather a human readable description of the instance. Finally, the third statement shows an example how to retrieve properties' values that have to be queried via joins.

5. Application for proposed Use Cases

Motivated by our use cases data quality mining and interoperability, we have presented our proposed architecture for encapsulating legacy database management systems and mapping them to a domain ontology. We will now go into further detail how this architecture provides advantages for those use cases.

5.1 Advantages for DQM

With the help of a more flexible data schema and the distinct naming of instances in the data management system, it is possible to annotate data directly taking data quality aspects into account without altering the underlying legacy data schema. A data quality ontology is designed (Grüning, 2006), its namespace is imported to the RDF repository and statements about instances of the original data management are made using the semantics of the imported ontology. Through this method, it is possible to separate the productive data management from the metadata management that holds statements about the quality of the data. The RDF repository can be distinct from the other data management systems.

The statements about data quality can be used to rate the outcomes of the algorithms for e.g. dispatching power plants or making predictions about the future developments of e.g. demand or supply of electrical energy. The vaguer the quality of the data is, the more questionable the outcomes of such algorithms are. This information can be used to broaden the safety margins of power plant resource planning in case of doubtful input data.

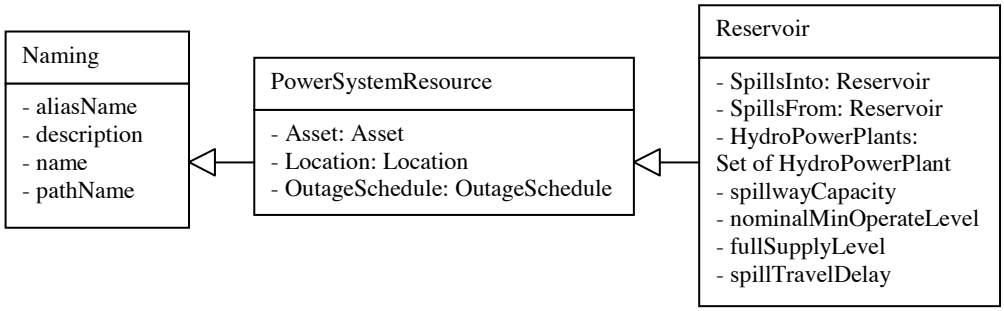

Figure 4.3.2: Extract of domain ontology and mapping target

5.2 Application for Interoperability

The integration of legacy databases using our architectural approach is the most striking application regarding interoperability issues. We have to distinguish mainly between two major issues: the usage of legacy data for messaging between applications and the extraction of legacy data for the purpose of CIM export e.g. for topology models. Both can be facilitated using the CIM model and the D2R approach, we focus on the latter case.

```
map:Reservoir a d2rq:ClassMap;
d2rq:dataStorage map:database;
d2rq:uriPattern "Reservoir/@@Reservoir.name@@";
d2rq:class vocab:Reservoir.
map:Reservoir__label a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:Reservoir;
d2rq:property rdfs:label;
d2rq:pattern "Reservoir#@@Reservoir.name@@".
map:Reservoir_name a d2rq:PropertyBridge;
d2rq:belongsToClassMap map:Reservoir;
d2rq:property vocab:Reservoir_name;
d2rq:refersToClassMap map:PowerSystemResource;
d2rq:join "Reservoir.name=PowerSystemResource.name".
```

Figure 4.3.3: Example of a mapping from legacy database scheme to a domain ontology

As depicted in Figure 4.2.1, we have exchanges between different companies agreeing on the same semantics. Schulte et al. (2006) have outlined the difficulties of using relational databases for creating power grid topologies using the CIM standard. The RDF data is part of the standard and the overall work amount implementing it using HP Jena and a relational database is enormous. Luckily, using an RDF repository simplifies the job - however, you need your data to be queried by SPARQL. The proposed architecture solves the query problem by wrapping the existing databases. It is feasible to use the existing sources and create standard RDF documents using fast and common SPARQL queries.

6. Conclusions

We proposed an architecture for encapsulating legacy database management systems and providing an ontological view on their data. The usefulness of the approach can be seen in two use cases: On the one

hand, the ontological view makes it possible to use the expressiveness of other ontologies through the flexible data schema and to make statements about the data stored in the original system. In the presented use case, information about data quality aspects of the data was used to decide about the reliability of the outcome of resource planning and prediction algorithms. On the other hand, a domain ontology, namely CIM, maintained in a standardization process, is used for semantic mapping to create interoperability between enterprises in the utility domain. As this domain is in a rapid change, where more and different market participants need to communicate with each other, a standardized communication is of high relevance. A quick adjustment of the enterprises' ICT systems to the standard is critical for successfully acting in this market.

Bibliography

Bizer, C., Seaborne, A. (2004): D2RQ -Treating Non-RDF Databases as Virtual RDF Graphs, 3rd International Semantic Web Conference (ISWC2004), Hiroshima, Japan.

Grüning, F. (2006): Data Quality Mining in Ontologies for Utilities. In: Klaus Tochtermann, Arno Scharl (Ed.): Managing Environmental Knowledge, 20th International Conference on Informatics for Environmental Protection: Proceedings, Shaker Verlag, Aachen 2006, p. 501-504.

Grüning, F. (2007): Data Quality Mining: Employing Classifiers for Assuring consistent Datasets. In: Jorge Marx Gómez, Michael Sonnenschein, Martin Müller, Heinz Welsch, Claus Rautenstrauch (Eds.): Information Technologies in Environmental Engineering 2007, Springer, pp. 85 - 94.

IEC (2003): IEC 61970-301: Energy Management System Application Program Interface (EMS-API) – Part 301: Common Information Model (CIM) Base, International Electrotechnical Commission.

Kampman, A.; Broekstra, J. (2007): Sesame, an open source RDF framework with support for RDF Schema inferencing and querying, http://www.openrdf.org/ (2007-05-14).

Martens, A. (2007): Konzepte und Technologien zur Abbildung von relationalen Datenbanken auf Ontologien, Ausarbeitung im Rahmen des Seminar Energiemanagement im WiSe 2006/2007, University of Oldenburg.

McMorran, A. et al. (2004): Translating CIM XML Power System Data to a Proprietary Format for System Simulation, IEEE Transactions on Power Systems, 1: 229-235.

Uslar, M. (2006): The Common Information Model for Utilities: An Introduction and Outlook on Future Applications, in: Rainer Eckstein, Robert Tolksdorf (Eds.): Proceedings of the XML days 2006 Berlin, XML-clearinghouse.de, p.135-148.

Prud'hommeaux, E.; Seaborne, A. (2006): SPARQL Query Language for RDF, W3C Working Draft 4 October 2006, http://www.w3.org/TR/rdf-sparql-query/ (2007-05-14).

Robinson, G. (2002) Key Standards for Utility Enterprise Application Integration (EAI), Proceedings of the Distributech 2002 in Miami, Pennwell Publishing.

Schulte, S. et al. (2007): Implementing and Evaluating the Common Information Model in a Relational and RDF-based Database. In: Jorge Marx Gómez, Michael Sonnenschein, Martin Müller, Heinz Welsch, Claus Rautenstrauch (Hrsg.): Information Technologies in Environmental Engineering - ITEE 2007 - Third International ICSC Symposium, Springer Verlag, Heidelberg, pp. 109-118,

W3Ca (2004): RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation 10 February 2004, http://www.w3.org/TR/rdf-schema/ (2007-05-14).

W3Cb (2004): Resource Description Framework (RDF): Concepts and abstract Syntax, W3C Recommendation 10 February 2004, http://www.w3.org/TR/rdf-concepts/ (2007-05-14)