SBVR as a Semantic Hub for Integration of Heterogeneous Systems

- A Case Study and Experience Report -

Ling Shi[1, 2], Dumitru Roman[3], and Arne J. Berre[3]

[1]Statsbygg, Pb. 8106 Dep, 0032 Oslo, Norway
[2]University of Oslo, Pb. 1072 Blindern, 0316 Oslo, Norway
ling.shi@statsbygg.no
[3]SINTEF, Pb. 124 Blindern, 0314 Oslo, Norway
{dumitru.roman,arne.j.berre}@sintef.no

Abstract. Extracting integration rules to handle semantic heterogeneity is one of the main challenges of achieving seamless connectivity between distributed systems. Semantics of Business Vocabulary and Rules (SBVR)'s machine and human readability and platform independence make it potentially suitable and interesting to study, as a central semantic hub of different systems. Semantic heterogeneity can be identified by comparing and analyzing vocabularies, fact models and business rules in the hub. Integration rules can then be extracted based on the semantic heterogeneity analysis. This article investigates and evaluates the usage of SBVR in heterogeneous systems integration. It provides a real-life case study and experience report on extracting integration rules based on an analysis of two Norwegian public sector's heterogeneous IT-systems modeled in SBVR.

Keywords: Integration rules, SBVR, Vocabulary, Fact models, Rules, Heterogeneity, Semantic heterogeneity, Ontology mismatch

1 Introduction

The amount of data has been exploding in the last decades and therefore also the need for integrated information from distributed systems. There are numerous data sources available in organizational databases and on public information systems[1]. A typical integration scenario is that two heterogeneous systems A and B are built for different business purposes for different users at different times by different software developers using different information models. The two systems often have heterogeneous semantics, i.e. vocabularies, data structures and business rules are different. One of the main challenges of achieving seamless connectivity of related information from the different source systems is to extract the integration rules that can handle the heterogeneity of the source systems. Integration rules cover mainly what parts of re-

[1] http://logic.stanford.edu/dataintegration/chapters/preface.html

sources and properties from different source systems could be integrated, under what conditions, and how the transformations should be formed.

Extraction of integration rules is based on the analysis of the heterogeneity of the different systems. Therefore the quality of the heterogeneity description decides directly the quality of integration rules. In order to describe the heterogeneity, semantics of the source systems need to be described first. Machine readable semantic models are preferred compared to only human understandable semantic models because machine readability provides the possibility of utilizing automated reasoning and thereby the possibility of automation of the integration rules extraction. On the other hand, human readable semantic models make it easy to involve and interact with domain experts and decision makers. The domain experts can validate the semantics of the existing models and their documentation. The domain experts' involvement in capturing semantics is essential, especially when little existing semantic information could be found from the existing sources. A machine and human readable language is therefore preferred to model both the semantics of the systems and the heterogeneity between them.

Semantics of data models can be expressed in various forms ranging from schemas to system documentation, by annotation to ontologies, etc. Some well-known information modeling languages are evaluated below for the criteria of machine and human readability. XML provides structure but relies on e.g. schemas to provide semantics and the tree structure of XML is not always suitable for capturing various types of semantic relations. Ontology models support semantic integration by the machine readable meaning of terms [1]. However, ontology models are often not understandable by people without ontology training and lack the ability to address business rules and integration rules which are crucial to the integration challenges addressed in this paper. The Unified Modeling Language[2] (UML) is commonly used to model vocabularies and rules with the help of UML profiles, but its suitability for representing knowledge in a way that is easy to understand for non IT skilled users is questionable. Semantics of Business Vocabulary and Rules (SBVR) [5] is both machine readable and structured and human understandable. The date-time vocabulary[3] (DTV) is one of the SBVR examples of machine and human readable semantics. SBVR is potentially suitable as the modeling language for the central semantic hub of different systems.

Figure 1 below illustrates an SBVR-enhanced integration hub in the context of integration of system A and B which is a typical problem in practice as shown with a real-life example in Sections 2 and 3. The different source models as database schema (DB), UML, Web Ontology Language[4] (OWL) and Resource Description Framework[5] (RDF), together with domain experts' knowledge and system documentation, etc., from both source systems provide input to the central integration hub to be expressed as vocabularies, fact models and business rules. A transformation between SBVR and well-known modeling languages such as the OWL has been documented

2 http://www.omg.org/gettingstarted/what_is_uml.htm
3 http://www.omg.org/spec/DTV/
4 http://www.w3.org/TR/owl-features/
5 http://www.w3.org/RDF/

in [2], and a transformation between SBVR and UML has been documented in [3, 7]. Heterogeneity can be identified by comparing and analyzing the SBVR models in the hub. The integration rules can be extracted based on the analysis result of heterogeneity.

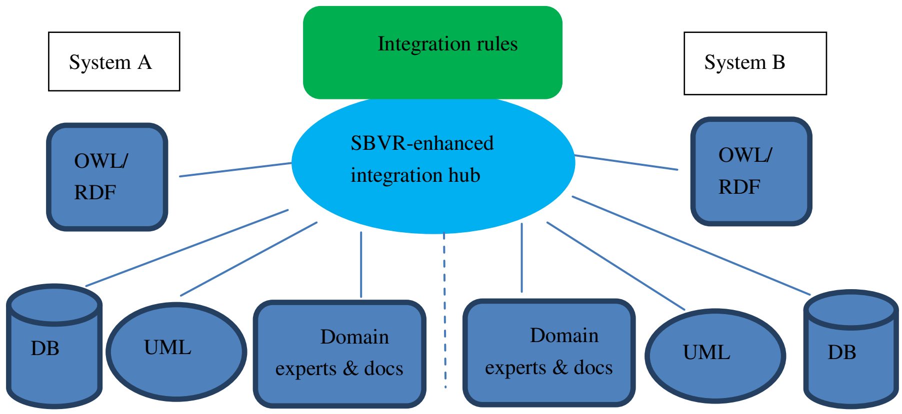

Figure 1. SBVR-enhanced integration hub

The rest of the paper is organized as follows. Section 2 introduces a case study with focus on the integration challenges of two IT systems. Section 3 presents the realization of SBVR-enhanced integration hub in the case study. It first introduces the overall SBVR-based approach to address the identified challenges, and then reports on the realization of the hub in the case study presenting the transformation of source models to SBVR vocabularies, fact models and business rules, and extraction of the integration rules based on the heterogeneity analysis. Finally, Section 4 summarizes the work that has been done, draws lessons learned in using SVBR as a semantic integration hub, and suggests directions for future work.

2 Case Study

Statsbygg (SB) is the Norwegian government's key advisor in construction and property affairs, building commissioner, property manager and property developer[6]. This public sector administration company has more than 800 employees and manages approximately 2.7 million square meters of floor space[7]. Many of SB's business areas are dependent on up-to-date building information such as the buildings' name, address, status, tenant information, etc. The property management system is called Propman. Propman registers, among others, buildings and building related informa-

---

6 http://www.statsbygg.no/System/Topp-menyvalg/English/
7 http://www.statsbygg.no/FilSystem/files/omstatsbygg/aarsrapport/SBaarsmelding2011.pdf

tion. Propman is client-server based and is built upon an Oracle relational database[8]. The available sources for building semantic models are database schemas, original system documentations and input from the domain experts.

The Norwegian national cadastral system called Matrikkel[9] stores the official data of buildings and building related information such as the buildings' name, usage area, status, address, land owner information, neighbor list and historical building changes. The Norwegian national map office is responsible for Matrikkel and it serves the data both as downloadable maps and map Web services. The available sources for building semantic models are class diagrams, a vocabulary and the Web services interfaces described with the Web Service Definition Language (WSDL).

Integration of the two systems is always a highly prioritized task among users and leaders in SB though there are integration difficulties, mostly at business and application layers. Building information such as address, status, municipal code, etc., is maintained manually in Propman though the information also exists in Matrikkel. Matrikkel is maintained frequently by municipal offices and other information providers. It often occurs that those types of data are out of date in Propman. One of the suggestions in SB to address the heterogeneity between the two aforementioned systems was to standardize the Propman with Matrikkel's building id and data structure for buildings. However, the heterogeneity of the two systems is inevitable since the two systems were originally built for different purposes and serve different users and customer groups. For example, SB administrates several properties abroad which are not registered in the Norwegian national Matrikkel. In such situations, standardized data structures do not guarantee interoperability at the scale expected by SB. Other examples of heterogeneity are shown in Section 3.

SB tried to update Propman data with building information from Matrikkel twice. In 2007 SB updated its building data based on the matching Matrikkel's building number in Propman though 25% buildings in Propman did not have the key registered. The quality of registered Matrikkel's building numbers was not good either. Some numbers were used several times on different buildings in Propman; some numbers did not belong to the right building. There were several kinds of errors after the updating process according to the feedback from the property managers. The second data updating was done in a half automated and half-manual way in 2012 and it tried to cover all the buildings including those without a Matrikkel's building number. Mismatches of data in the two systems were listed up and suggestions of modifications were delivered to the property management administrator who forwarded them to each property manager individually. Major reasons for this time and resource consuming process included the lack of a suitable vocabulary that could define terms like building, building's built area, building's address, etc., and also the lack of clear structures and rules that could be used to decide which building should be included in the register and what to do if mismatches between Matrikkel and Propman occurred. Each property manager then made an evaluation based on his/her domain knowledge and the corrections were therefore not 100% consistent.

---

8 http://www.oracle.com/us/products/database/overview/index.html

9 https://www.matrikkel.no

Neither of the integration approaches provided expected results and they were either time or resource consuming or error-prone. SB was looking for a more effective and systematic approach that could improve the data quality in Propman. The heterogeneity should be concretized and classified, and then the integration rules should be extracted to handle the heterogeneity problems with standardized processes, and to avoid the inconsistency caused by the missing rules. To achieve this goal, an approach has been developed as shown later in Section 3.1.

3 Realization of the SBVR Hub in the Case Study

This section presents a real-life example of implementing the SBVR-enhanced integration hub for the two aforementioned heterogeneous systems. An overall approach is introduced first, and then the step by step realization of the approach is presented.

3.1 Overall Approach

The case was simplified to concentrate on the integration of buildings and related basic building information, though there are plenty of other kinds of building information that could be integrated such as addresses, owners, or land. The suggested approach focuses on the SBVR-enhanced integration hub and the extraction of integration rules based on a heterogeneity analysis. It is designed to include the following parts and steps:

Part 1: Establishing the SBVR hub: Building the vocabularies, fact models, and business rules for the source systems.

Part 2: Extracting Integration Rules: Identifying the mismatches of term definitions in the vocabulary, business rules and fact models; Building the integration rules after the mismatches.

The steps can be automated if proper tools are available, however manually approaches can also be used if necessary.

3.2 Building the Vocabularies

A template to define the data fields in the vocabulary was agreed upon first, and then two sets of SBVR vocabularies were established to separate the two system domains, the Matrikkel system and Propman respectively.

Vocabulary Template

SB has a cross domain working group called master data group whose main task is to define and maintain master data of the company. Domain experts from different business areas are invited to the group to work on their domain related terms and give input to the definitions. The group extended a template[10] from the Norwegian Agency

[10] http://standard.difi.no/filearchive/2012-05-13-mal-begrepsbeskrivelser-1-0.pdf

for Public Management and eGovernment (Difi)[11] to define the data fields included in the vocabulary as shown below.

- Difi fields:
  - Difi mandatory: Identifier, Recommended Term, Definition, Source, Discipline, Effective from, Responsible, Language, Expires, Reference to versions, Classification;
  - Difi recommended extension: Related concepts, Comment;
  - Difi administration documentation: Contact, Last Modified, Modified By, Status.
- SB extension fields: Synonym, System, Category, Approval Date.

The Matrikkel Vocabulary

The most challenging part of building a vocabulary was to reach a common sense on the definition in the organization or company. The Matrikkel system has already done part of the work and a definition page is provided on their website[12] with total of 92 terms dated September 1st 2012. Below is the original definition of the term "Building" in Matrikkel from the website.

Table 1. The original definition of the term Building in the Matrikkel system

Includes buildings and building changes. The basis for the definition of buildings is Eurostat's definition: "Can be used separately listed for a permanent purpose and are suitable or intended to protect people, animals or things." GAB system and DEK were two predecessors of the Matrikkel system. GAB system has from the beginning been associated with the Norwegian Standard NS 3940 Area and volume calculation of buildings as the basis for calculating area of buildings. This principle continued in the Matrikkel. The area concept to be used in Matrikkel is called usage area. It is described in the standard and in the Matrikkel instructions. One precondition for a valid usage area is that the building meets the requirements of the standard of measurability. Based on these rules, carports (open garage), which only includes open space, fall outside of the definition. The same applies to tank constructions since such constructions are not normally accessible via doors or the like. It indirectly implies that buildings in Matrikkel have usage area. It is however still open to register buildings that are otherwise not subject to registration. The area should then not be recognized as usage area but as an alternative type of land use. All buildings with a usage area of 15 m² or more shall be recorded in Matrikkel, but smaller buildings can also be registered.

The above definition includes not only the definition of the term, but also the other valuable information as input to the vocabulary, fact model and business rules. The table below shows the original definition's input to the vocabulary template as described at the beginning of this section.

---

[11] http://www.difi.no

[12] http://www.statkart.no/filestore/Matrikkelavdelingen/Foeringsinstruks/chapter02.html (in Norwegian)

Table 2. Applying the vocabulary template to the term Building in Matrikkel

Vocabulary Field	**Value**
Identifier	M.Building
Recommended term	Building
Definition	Includes buildings and building changes. The basis for the definition of buildings is Eurostat's definition: "Can be used separately listed for a permanent purpose and are suitable or intended to protect people, animals or things."
Source	Eurostat, NS3940
Responsible	The Norwegian National Map Office
Related concepts	Usage area

The first sentence of the original definition indicates the term's relation to buildings and building changes. The original definition text also states a relation between Building and its attribute usage area. Both relations should be taken care of in building the fact models. Several business rules on what could be registered as a building in Matrikkel are included in the original definition text and they should be taken care of in building the business rules.

The SB Vocabulary

The master data group in SB arranged workshops with domain experts to collect the company's central terms and definitions. Some terms are cross-domain terms and domain experts had different understanding, focus and usage of the terms. The master data group worked as a negotiator in order to reach an agreement between the domain experts. When disagreement could not be resolved at this level, it had to be escalated to the higher administration level to reach a final decision. The result was an Excel sheet with more than 500 terms with names, definitions and related information.

For example, the definition of Building in SB is as follows:

Table 3. The original definition of the term Building in the Propman system

A building is a continuous building mass. In many cases, larger building masses can be defined as several buildings because the original building has been extended, and each gets its building number and building name. This is due to the preservation of historical data, due to different building status: e.g. building under construction, or has just been completed.

Table 4. Applying the vocabulary template to the term Building in the Propman system

Vocabulary Field	**Value**
Identifier	SB.Building
Recommended term	Building
Definition	A building is a continuous building mass.
Source	Propman's system documentation
Responsible	SB
Related concepts	

The above definition indicates that a normal criterion to identify a building with a unique building number is the continuous building mass. However, exceptions are allowed when a building is extended or under special conditions. These could be used as inputs to build the business rules.

3.3 Building the Fact Models

The Matrikkel system had UML models in its documentation and the database schemas of Propman were accessible. Those sources were transformed to SBVR fact models as described below.

Matrikkel - Transformation from UML to Fact Models
UML class diagrams could be transformed to fact models using some intuitive transformation rules based on ideas from [7, 8].

A single UML class may include name, attributes and operations [4]. The class name could be transformed to a term name in the fact model. The class attributes could be transformed to properties, one of the four special-purpose element of structure as described in [5]. The wording for property is "has". There is no direct transformation from UML class operations to fact models though the operations could be analyzed and implemented as rules if applicable. The interpretation of operations to rules is not covered in this study.

The generalization between UML classes could be transformed to categorization, also one of the four special-purpose element of structure in fact models. The wording for categorization is "is a category of". The generalization could also be transformed to classification if the subclass is an instance of the superclass, e.g., subclass Canada vs. superclass Country. The wording for classification is "is classified as".

The composition could be transformed to composition in fact models, the wording for composition is "is composed of" or "is included in". Other type of associations could be transformed to fact types.

Due to the scope of the case study, the transformation rules used in the case study do not cover the transformation of, e.g., data types of attributes, multiplicity of attributes, multiplicity of associations, and aggregation in UML classes to fact models.

Figure 2 shows an example of a fact model modeled in a tool called FactXpress[13]. The model is generated from the UML model of building[14] in the Matrikkel system based on the transformation rules specified above. The text in parentheses is the original text in Norwegian.

---

13 http://www.rulearts.com/FactXpress
14 https://www.test.matrikkel.no/matrikkel/docs/Domenemodell.html#analysemodellen_bygg

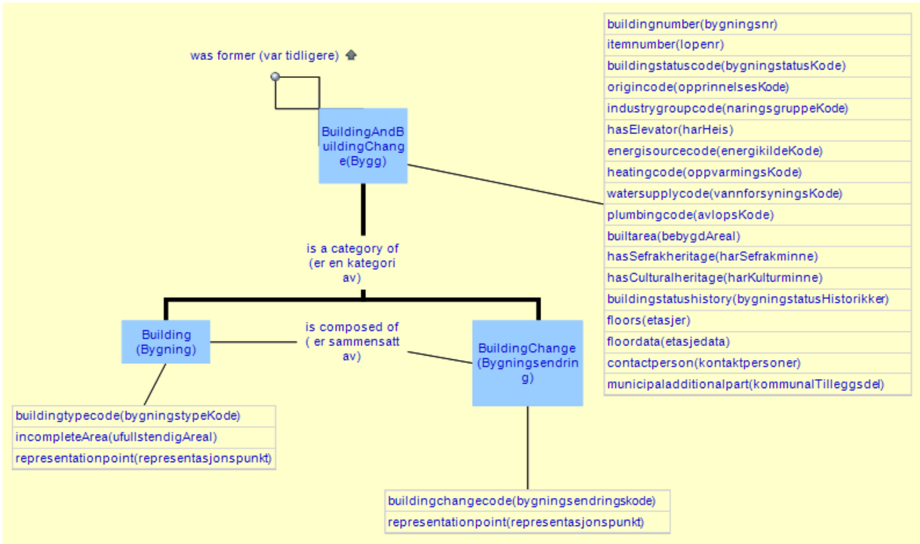

Figure 2. An SBVR fact model generated from a Matrikkel UML class diagram

Propman - Transformation from Database Schemas to Fact Models

The database schemas and system documentations were two of Propman's accessible source for semantics. Class diagrams were used as the bridge in transforming database schemas to fact models since transformation between database schemas and class diagrams can be automated by several database modeling tools. The database schemas were first imported to a class diagram using the existing reengineering tool in EA[15]. The legacy system Propman had not built relationships between the tables; thereby there were no associations between the classes. The information retrievable from the class diagram was therefore limited. There were other sources though, for example the system documentations. The statements that were helpful from the system documentations were, e.g., "One Complex includes one or more lands" and "Each land could build one or more buildings". The figure below shows the fact model modeled based on the database schema generated class diagram and system documentations.

[15] http://www.sparxsystems.com/products/ea/index.html

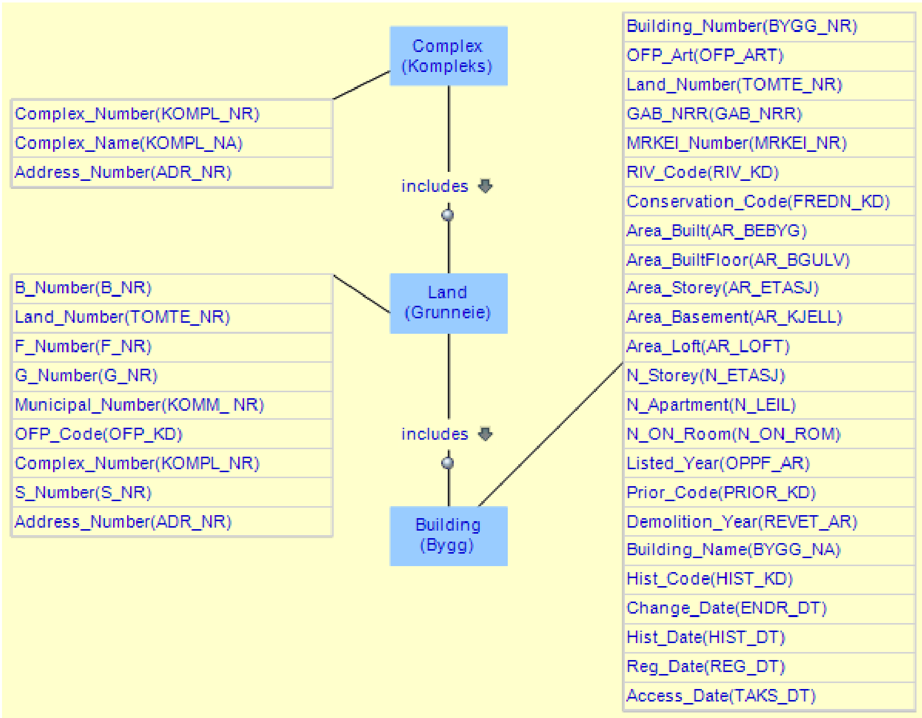

Figure 3. The fact model of SB Building

Comparing Fact Models and Vocabularies

The fact models enrich the definitions in the vocabularies built as outlined in Section 3.2 by adding attributes of a concept and relations to other concepts. Moreover, fact models can also be used to validate the definitions from Section 3.2.

A common challenge in integration is using the same term for different meanings. A fact model will help with the identification of this kind of misleading information by showing attributes and relations of the concepts in a structured way. For example, the building definition in Table 2 states that a building "Includes buildings and building changes". The fact model in Figure 2 shows three concepts, i.e. "BuildingAndBuildingChange", "Building" and "BuildingChange". The definition in the vocabulary should then be modified to reflect the fact models.

3.4 Building the Business Rules

The Matrikkel Business Rules

Some of the definitions in Matrikkel's vocabulary also include rules. Those rules could be extracted from the textual definition, for example:

R-M-bygg-4:

Each building with usage area 15 m² or larger should be registered in Matrikkel system. Smaller buildings can also be registered, but is not compulsory.

The SB Business Rules

The rules in SB were collected by interviewing the users and sending inquiries to responsible Propman system administrators. Here are some examples.

R-P-bygg-1:

SB registers both buildings larger than and smaller than 15 m^2

R-P-bygg-2:

Each matriculated building in SB should be registered with a Matrikkel's building number.

3.5 Identifying the Mismatches of Vocabulary, Structures and Business Rules

This section compares the vocabulary, structures and business rules of the two systems in the SBVR hub, and identifies the mismatches. Heterogeneity on resources usually could be confirmed already by comparing the term definitions in the two vocabularies. Further analysis could be done on comparing properties and fact types of each resource in the two fact models. Finally, the business rules defining the scopes or other attributes of resources could be compared to identify, e.g., the difference of the scopes.

The Vocabulary

Comparing the definition of a term in the vocabularies provided direct indication whether the terms were identical or not. The sources of the term definitions were compared first and found out to be different for the term Building in the two systems. Then the definition texts were compared, Matrikkel's definition focused on a building's primary functions and SB's definition focused on the physical building mass. The term Building was therefore not identically defined in the two systems which means further analysis was necessary to identify the mismatches on the structure and business rules levels.

The Structures

Fact models and their visualization provide an intuitive way of analyzing the structures. The properties and fact types of each resource in the two models were considered. One of the heterogeneity types is terminological heterogeneity [6]. It occurs when two properties with different names deal with the same information. For example, the builtArea is a data property of Building in Matrikkel, while Area_Built is a property of Building in SB. This is a possible source of information duplication and mismatching. The integration rule should therefore address the issue by defining which system is the original source of the data property. For example, in this case, SB has the responsibility to report the Area_Built value to the Matrikkel system. Therefore SB's Area_Built is the original source of this data property.

The Business Rules

Comparing the business rules of a specified term provided further information on semantic similarities and differences. The example below shows the different policies on what should be registered as Building by Matrikkel and SB. Those rules mismatch is caused by the heterogeneity in coverage of the resource Building. The heterogeneity in coverage is a subtype of conceptual heterogeneity as classified in [6]. The difference should be handled later in the integration rules.

3.6 Building the Integration Rules

This paper covers terminological and conceptual heterogeneity though other types of heterogeneity should be included in the identification and classification work in Section 3.5. The integration rules discussed below addresses the two types of heterogeneity.

An integration rule can be formed using a template like this: *What part of resource/property X in source system A should integrate with what part of resource/property Y in source system B with which integration keys under which conditions?*

To handle the terminological heterogeneity, an integration rule should define the original source of a data property in case of duplications. For example the following integration rule could solve this type of heterogeneity. This kind of integration rule can be interpreted as an ontology rule using, e.g., rdfs:subPropertyOf or owl:sameAs. The study of interpreting integration rules to ontology rules falls out of the scope of this paper.

R-Int-bygg-1:

A Building's property Area_Built in Propman is the same as a Building's property Built_Area in Matrikkel with the integration key "Matrikkel's Building number".

One alternative way to handle the "Difference in coverage" type of conceptual heterogeneity is to define the overlapping part as integration part. For example, the Building in Matrikkel and the Building in Propman are overlapping when both have the same Matrikkel's building number, then the integration rule below defines that only building with Matrikkel's building number can be integrated.

R-Int-bygg-2:

Each building with Matrikkel's buildingnummer in SB can integrate with a building in Matrikkel on Matrikkel's building number.

4 Conclusion and Further Work

The approach presented in this paper focused on extracting robust integration rules based on the heterogeneity analysis in an SBVR integration hub. Semantics of source systems could be modeled in or transformed to SBVR, and heterogeneity of different source systems could be identified by comparing those elements in the hub. Integration rules could then be extracted based on the identified heterogeneity. The information in the hub was understandable directly to both domain experts and IT personals. In this way, domain experts were able to follow the process all the way and their interaction with integration software developers was faster and more effective since they used the same mechanisms for communication. This also made software documentation easier since most of the explanations on why and how the integration rules are extracted was already documented in the process in a human understandable language.

Currently, the transformation from source systems to SBVR hub is manually done and information could be lost in the transformation process. The identification of heterogeneity is also manually done, and so is the classification of heterogeneity types. The extracting of integration rules is based on human reasoning and still needs significant involvement of domain experts.

The case study presented in this paper is a preliminary attempt to design and implement an integration analysis framework based on machine and human readable SBVR. The approach in this paper can be further complemented and extended as follows. A transformation from source models to SBVR models needs to be automated to simplify the process of building SBVR-enhanced integration hub. A deeper review of the existing functionality of available tools should also be part of the further work. Comparing and analyzing heterogeneity needs to be standardized and automated if possible to reduce the inconsistency caused by human involvement. A state of art study of classification of ontology mismatches and their representation in SBVR would be natural to be done. Extraction of integration rules based on heterogeneity should be designed as an ontology reasoning process, where applicable, to reduce the inconsistency caused by human involvement. Other rule modeling languages such as Rule Markup Language (RuleML) could be evaluated for automated generation of integration rules.

Acknowledgment. The work of Dumitru Roman and Arne J. Berre is partly funded through the Semicolon II, PlanetData, and BigFut projects.

References

1. Uschold, M., Gruninger, M.: Ontologies and Semantics for Seamless Connectivity (2004)
2. Karpovic, J., Nemuraite, L.: Transforming SBVR Semantics into Web Ontology Language OWL2: Main Concepts. In: Proc. 17th International Conference on Information and Software Technologies IT 2011 (2011)
3. Nemuraite, L., Skersys, T., Sukys, A., Sinkevicius, E., Ablonskis, L.: VETIS tool for editing and transforming SBVR business vocabularies and business rules into UML&OCL models. In: Information Technologies, IT 2010, Kaunas, Lithauania, April 21-23, 2010, 377-384 (2010)
4. Fowler, M., Scott, K. : UML Distilled Second Edition (2000)
5. Ross, R. G.: Business Rule Concepts, third edition (2009)
6. Euzenat, J., Shvaiko, P.: Ontology Matching (2007)
7. Cabot, J., Pau, R., Raventós, R.: From uml/ocl to sbvr specifications: a challenging transformation. In: Information Systems. Elsevier, Amsterdam (2009)
8. "Semantics of Business Vocabulary and Business Rules (SBVR), version 1.0, v1.0.0 Object Manangement Group (OMG), Jan. 2008. [online] [last accessed: April. 2012]. http://www.omg.org/spec/SBVR/1.0/PDF/