A Study in User-centric Data Integration

Heiner Stuckenschmidt[1,2], Jan Noessner[1] and Faraz Fallahi[3]
[1]*School of Business Informatics and Mathematics, University of Mannheim, 68159 Mannheim, Germany*
[2]*Institute for Enterprise Systems (InES), L 15, 1-6, 68131 Mannheim, Germany*
[3]*ontoprise GmbH, An der RaumFabrik 33a, 76227 Karlsruhe, Germany*

Keywords: Data Integration, User Study, Software Tools, Usability.

Abstract: Data integration is a central problem in information systems. While the problem of data integration has been studied intensively from a technical point of view, less attention has been paid to user aspects of data integration. In this work, we present a user-centric approach to data integration that supports the user in finding and validating mapping rules between heterogeneous data sources. The results of our report underline that the user-centric approach leads to better integration results and is perceived as being more intuitive, especially for users with little or no technical knowledge.

1 INTRODUCTION

The problem of information integration is omnipresent in information systems and can be seen as one of the major challenges, both, on the technical and the organizational level. In this paper, we focus on the problem of transferring complex data from one into another representation in order to support exchange between different systems, also referred to as data integration.

The problem of data integration has been studied intensively on a technical level in different areas of computer science (Halevy et al., 2006; Euzenat and Shvaiko, 2007). Researchers have investigated the automatic identification of semantic relations between different datasets (Euzenat and Shvaiko, 2007) as well as the representation and use of identified relations for data transfer and query answering (Bellahsene et al., 2011). A prominent line of research investigates the use of ontologies - formal representations of the conceptual structure of an application domain - as a basis for both, identifying and using semantic relations.

In contrast to this work, we are more interested in data integration as a task and in how we can empower the user to solve this task more efficiently and effectively. A successful solution to this problem would have significant implications for data integration in industrial practice. Traditionally, data integration is done by computer science experts of an enterprise or even outsourced to a service provider specialized in solving data integration problems. The fundamental problem of this approach is the fact, that the integration experts are often not experts with respect to the data that is to be integrated. This means that their ability to identify conceptual errors within the integrated data is limited. As a consequence, errors are often found by the user when the data has already been migrated. Fixing such problems at this point typically requires intensive communication between the user and the integration expert and causes overhead. This efficiency loss could be avoided if the user, who knows the data, but not necessarily the underlying technology, would be able to identify and fix integration problems directly.

Following the design science approach (Von Alan et al., 2004), we designed and implemented a user-centric data integration tool called MappingAssistant(Noessner et al., 2011; Fallahi et al., 2011)[1]. This tool allows the user to specify and validate semantic mappings between two datasets following an interactive process model: after specifying a semantic mapping, the system automatically translates data using the specified mapping and presents selected results of this translation to the user, who can then mark individual results as correct or incorrect. Based on this user input, the system identifies mistakes in the semantic mappings by asking the user about the correctness of

[1]The interested reader is referred to http://www.ontoprise.de/de/forschung-und-entwicklung/mappingassistant/ for further information. An illustrative video is available at http://www.youtube.com/watch?v=72abBBTfl_E.

Stuckenschmidt H., Noessner J. and Fallahi F..
A Study in User-centric Data Integration.
DOI: 10.5220/0003971700050014
In *Proceedings of the 14th International Conference on Enterprise Information Systems* (ICEIS-2012), pages 5-14
ISBN: 978-989-8565-12-9

certain statements dealing with the conceptual model of the data.

In this paper we report the results of a user study that compared the user-centric integration method implemented in the MappingAssistant tool with the use of a pure mapping editor that does not interact with the user. We show that the use of the MappingAssistant approach significantly improves the performance of human users and that the method is especially suited for supporting users with few technical skills.

The paper is organized as follows. In Section 2 we discuss the concept of user-centric data integration and briefly review related work on the topic. Section 3 briefly introduces the MappingAssistant tool that we created. The user study evaluating the MappingAssistant tool, which is the main topic of this paper, is described in Section 4. We present our research hypotheses, define the experimental setting and the used datasets. Section 5 presents and discusses the results of the study. The paper closes with some conclusions in Section 6.

2 USER-CENTRIC DATA INTEGRATION

Studying data integration from a user point of view is a relatively new field of investigation. Traditionally, user studies have rather focused on the problem of personal information management. Data integration can be seen as an important aspect of personal information management (Teevan et al., 2006), however, research has focused on other aspects like the organization of emails or documents. In a recent study, Gass and Maedche have investigated the problem of data integration in the context of personal information management from a user-centric point of view (Gass and Maedche, 2011). The scenario addressed in their work, however, focuses on the integration of rather simple data schemas, in that case personal data where the task is mainly to map properties describing a person (e.g. name or bank account number). In many data integration scenarios like product data integration or computer aided design and manufacturing we face much more complex conceptual models and mapping rules.

Recently, researchers in ontology and schema matching have recognized the need for user support in aligning complex conceptual models (Falconer, 2009; Falconer and Storey, 2007). Most approaches are based on advanced visualization of the models to be integrated and the mappings created by the user (Granitzer et al., 2010). While the appropriate use of visualizations is known to be a key aspect for successful manual data integration, visualizations quickly reach their limits in the presence of very complex or very large models.

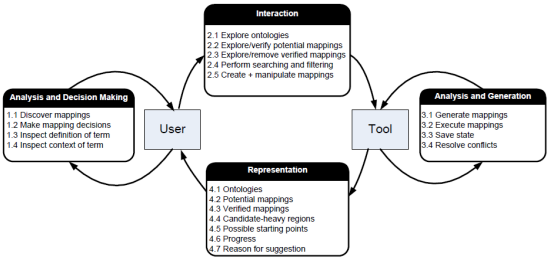

Figure 1: The cognitive support model for data integration by Falconer (Falconer and Noy, 2011).

As a result, recent work tries to go beyond pure visualization support and tries to include cognitively efficient interaction strategies to support the user (Falconer and Noy, 2011). Falconer proposes an interactive strategy for data integration where the integration task is distributed between the user and the tool (compare Figure 1).

3 THE MappingAssistant TOOL

In data integration much work has been invested in producing data integration rules with ontology matching systems automatically (Euzenat, J. and Ferrara, A. and Hollink, L. and Isaac, A. and Joslyn, C. and Malaisé, V. and Meilicke, C. and Nikolov, A. and Pane, J. and Sabou, M. and others, 2010). However, these rules are still error-prone and, therefore, need to be supervised by a human domain expert. This supervision should be effectively supported by applications.

Existing applications like AgreementMaker (Cruz et al., 2009) present the generated data integration rules directly to the user. Although these tools try to visualize complex data integration rules in an easy understandable way, the evaluation of these rules still requires a significant amount of expert knowledge. Furthermore, in real-world scenarios users are usually confronted with ill-labeled concepts, making the analyses even more complex and time-consuming.

The approach of MappingAssistant (Noessner et al., 2011; Fallahi et al., 2011) simplifies the alignment evaluation process by investigating the direct consequences of the data integration rules. In particular, the MappingAssistant approach allows to evaluate the instance data like product numbers or customers directly, rather than analyzing complex data integration rules. One of the advantages is that in real-world scenarios the domain expert often has sophisticated

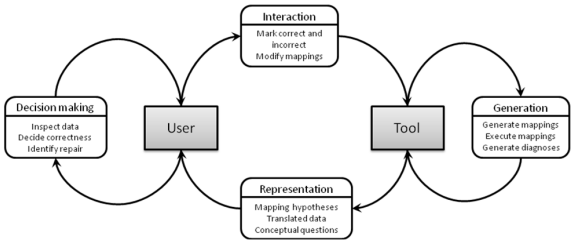

Figure 2: Modified cognitive support model implemented in the MappingAssistant Tool.

knowledge about the instance data in his domain.

The MappingAssistant approach implements the cognitive support model for data integration by Falconer (compare Figure 2). In the *decision making process*, the user inspects the data and decides which concept he wants to examine. In the example shown in Figure 5 the user selected FamilyCar in the target schema. In the *interaction* step, the user identifies those instances which have been classified incorrectly and marks them as correct or incorrect. Due to the amount of instances a user can be faced to diagnose we utilize different clustering techniques in order to reach data simplification. Attribute-driven combinations of weighted hierarchical and partial clustering algorithms, as mostly described in (Hair Jr et al., 1995), are therefore utilized. In our example of Figure 5 the `MX5_Mieta` is a two seated car and, thus, not a `FamilyCar`.

In the *generation* phase a diagnostic algorithm computes the minimal amount of user questions which are needed in order to determine wrong rules or facts. When the user depicts an instance as incorrect, we generate a proof-tree for the corresponding concept-assertion like `FamilyCar(MX5_Mieta)` in our example. Since the user evaluation is correct by assumption, the prolog-based proof-tree must contain at least one wrong node. In order to determine this wrong node our approach traverses the proof-tree in a way that the amount of user questions are minimized for correct as well as for incorrect answers of the user.

In the *representation* phase, the MappingAssistant tool generates questions based on the information it gets from the proof-tree algorithm before. These questions are represented to the user in natural language based sentences in a todo-list, as shown in Figure 5. If the wrong rule or fact already has been determined the algorithm terminates. Otherwise the user is faced with the next question, which was determined by the diangostic algorithm in the *generation* phase. The approach is implemented as an extension of the OntoStudio Ontology Engineering Workbench (Maier et al., 2003).

4 A USER STUDY

As part of the MappingAssistant project, we carried out a study in user-centric data integration. The goal of the study was to show that an interactive approach to data integration leads to better results than traditional approaches. In the following we discuss the goals, the design and the results of this study in more details. In Subsection 4.1 we first define the hypotheses tested in the study in more details, we then present and justify the experimental design in Subsection 4.2.1 and present the dataset used in the experiment in Subsection 4.3. Finally, Subsection 4.4 provides demographic information about our subjects.

4.1 Hypotheses

The user study was carried out to establish the general hypotheses of our work, which can be phrased as follows:
The cognitive support model helps users to correctly and efficiently integrate complex data.
We have to further substantiate this hypothesis in several ways to arrive at a useful experimental design. In particular, we have to be more explicit about the nature of the support model, the kind of users addressed as well as the integration task to be solved and the notions of efficiency and correctness. In the following, we thus provide more concrete definitions of the hypotheses to be tested.

4.1.1 Cognitive Support Model

We consider the extended cognitive support model in the way it is implemented in the MappingAssistant System (Figure 2): The system generates mapping hypotheses and executes them. Traditionally, automatic generation of such mappings are generated by lexical and/or tree structure based matching algorithms as described in (Euzenat and Shvaiko, 2007). The results are represented in terms of translated data instances. The user inspects the translated data and decides on the correctness of data items thereby providing feedback to the system. The user thereby triggers a second interaction cycle, where the system asks questions about the mappings and the underlying conceptual model waiting for the user to answer them.

4.1.2 Propective Users

The motivation for designing the extended cognitive support model and for implementing the MappingAssistant was to enable users with little or no technical knowledge in data integration. Thus, our refined hypothesis is that users with limited knowledge in conc-

eptual modeling and data integration show a better performance when supported by the cognitive support model. Further, we assume that the positive effect is stronger for people with very little knowledge than it is for users with more knowledge in these areas.

4.1.3 Integration Task

We decided to focus on the task of validating an existing set of mappings rather than creating a new set of mappings. If each subject is asked to *create a new* set of mapping, we would get many different solutions which might all be correct. Especially when complex data structures are involved the same integration task can be solved using different sets of mapping rules. This makes it extremely difficult to measure the correctness and completeness of the solution provided by a user. In contrast to this, identifying errors in an *existing* set of mapping rules is a more well-defined task that has a unique solution, provided the test data is designed in a suitable way.

4.1.4 Quality Criteria

We expect two positive effects of using the cognitive support model. The first one is efficiency, which means that a user is able to find errors in a set of mapping rules in a shorter period of time. The second one is effectiveness: we assume that a user is able to find more errors with the MappingAssistant approach, which would have remained undiscovered without the support by the system.

In the context of a controlled experiment, it is hard to distinguish these two effects; our hypothesis is that users are able to find more errors in a fixed period of time.

4.2 Experimental Design

In the following, we describe the experimental design of a user study we carried out to test our hypotheses.

4.2.1 Study Design

The general idea of the experiment is to compare the MappingAssistant approach to data integration with a conventional approach that is solely based on the use of a mapping editor. As a result, the study consists of two tasks, both concerned with identifying errors in a set of mapping rules that combines two conceptual schemas. In order to control external parameters of different integration tools, both tasks are carried out using the OntoStudio knowledge Engineering environment. OntoStudio contains a mapping editor that can be used to visually inspect and modify a set of mapping rules as well as an extension that implements the MappingAssistant approach.

Based on this technology we created the study design depicted in Figure 3.

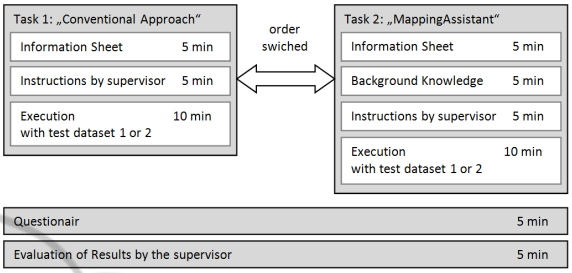

Figure 3: Design of the user study.

Participants have to solve two integration tasks in a timeframe of 10 minutes each. Before each task, the participants are briefed about the task. The information given to the user consists of an information sheet explaining the task and instructions by a supervisor who answers questions about the task without providing hints towards the expected results. The timeframe for this instruction phase has been determined individually for the two tasks, but is the same for all participants.

The order in which the two tasks are carried out is switched after every subject in order to avoid an efficiency bias for the second task due to a training effect. Since all participants perform both tasks, we do not need to divide the users in groups, but can compare the performance and experience of the users directly.

Furthermore, two different datasets are used, which are assigned to the two different tasks randomly for the purpose of ruling out a possible bias due to a different level of difficulty. After a subject has carried out both tasks, he or she is asked to fill in a questionnaire on the perceived difficulty and support by the tool as well as on the level of expertise of the subject.

Thus, in our study the *independent variable* is determined by either using the conventional approach (Integration Task 1) or the MappingAssistant approach (Integration Task 2). The *dependent variable* is the number of errors the subjects found in the respective dataset(Wohlin, 2000).

The following subsections provide more information about the two integration tasks, data and subjects used in the study as well as on the contents of the questionnaire.

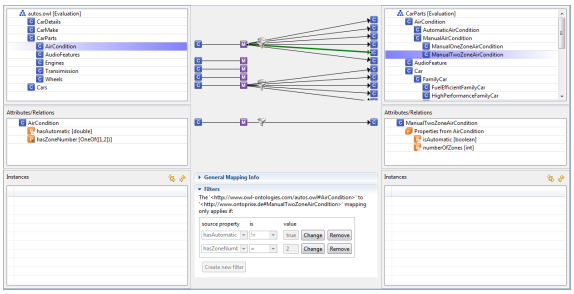

Figure 4: Traditional user interface for creating and evaluating semantic mappings.

4.2.2 Integration Task 1 (Conventional Approach)

Integration Task 1 consists of using the conventional user interface of the OntoStudio Mapping Editor to discover errors in a set of mapping rules. Figure 4 shows the user interface used in this task. It shows the conceptual model of the source data on the left and the source model on the right hand side. The models consist of classes, relations and instances.

The mapping rules connecting the two models are displayed visually in the middle of the screen. Below the visual representation of the mappings, specific filter conditions for mapping rules can be displayed in the form of pre-selected drop-down menus by clicking on a mapping rule.

The task of the user is to inspect the mapping rules for errors by navigating the conceptual models and the mapping rules and their filter conditions. This standard configuration of the tool does not provide special functionality for focusing on problematic issues, meaning that the user has to actively search for errors without being guided by the tool.

4.2.3 Integration Task 2 (MappingAssistant Approach)

The second task consists of solving the same problem, namely the identification of errors in a set of mapping rules. However, instead of the conventional mapping editor, the MappingAssistant plug-in is used. Its interface also shows the conceptual model of the source and the target data set and the mappings between the elements. Instead of the filter conditions of the rules, however, the interface shows results of translating data using the mapping rules, as well as a todo-list with questions generated by the tool that have to be answered by the user (compare Figure 5). Additionally, the plug-in allows for utilizing different clustering techniques in order to reach data simplification on the instance level.

The task of the user is again to inspect the mappings for errors. This time, however, an interactive process is used. The user actively selects a class in the target model and inspects the instances that have been created by executing the mapping rules. Based on his or her knowledge of the domain, the user can mark rows in the data table as incorrect indicating that they are not mapped coorectly to the selected class in the target model. His action then triggers a diagnosis procedure that generates yes/no questions about the conceptual structure of the data and displays the questions in the todo-list. The user has to answer these questions thereby guiding the semi-automatic diagnosis process to the errors. Once an error has been found, the user can select another class in the target schema and so forth.

4.2.4 Questionaire

The questionnaire that had to be filled in by all participants consisted of four parts. All questions except demographic questions had to be answered on a 1-5 scale:

1 definitely disagree,

2 rather disagree,

3 neither agree nor disagree,

4 rather agree,

5 definitely agree.

In the following, the different question categories are discussed:

Previous Knowledge: this category contained questions about the knowledge and expertise of the subject in the areas of data modeling and data integration. Examples of questions from this area were:

- I am experienced with using complex software tools
- I am used to apply filter rules for selecting data (e.g. in Microsoft Excel)
- I have good knowledge about formal data models
- I have good knowledge in data matching and integration

Task 1: This part of the questionnaire explicitly addresses the experiences of the subjects with respect to performing Task 1. The goal was to get a better idea of the perceived complexity and difficulty of this task. Examples of questions were:

- I was confused by the representation of the mapping rules

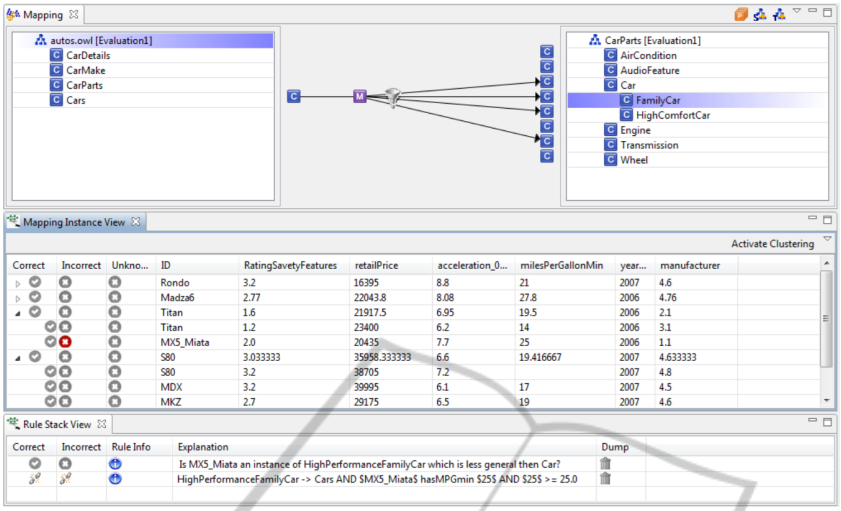

Figure 5: Interactive user interface of the MappingAssistant data integration tool.

- I was able to decide on the basis of the filter expressions whether a mapping is correct or not
- It was easy to work off the mappings without missing out on something

Task 2: This part focused on the experience of the subjects with solving task 2 and being supported by the MappingAssistant tool. The goal was to judge the level of support provided by the system. Example questions were:

- It was easy to identify wrong instance data
- The presentation of wrong mapping rules by the system was intuitive
- The attributes of translated data items helped me to identify mistakes

Comparison of Task 1 and Task 2: In order to be able to compare task 1 and task 2 the following question was asked in both tasks:

- The handling was intuitive

Demographics: Finally some demographic information was asked including age, gender and occupation of the subjects.

4.3 Datasets

When selecting the datasets for the study, we had to find a trade-off between the following issues. On the one hand having a problem that can be understood and solved within the limited timeframe of the study and on the other having enough complexity to adequately represent a realistic data integration challenge and supporting our assumption that our method works well for complex problems. In order to be able to satisfy these needs we decided to use a combination of existing data and data that has been created manually for the study.

4.3.1 Source Dataset

We decided to use a technical domain because of the typical rich feature-sets and complex definitions. As it turned out that real world datasets were much too large and complex to be handled in the study, we chose an instructional dataset from the car-selling domain[2] that was automatically converted to fit the data model of the OntoStudio Tool. The dataset contains 324 data records that are described using more than 100 attributes. In addition the data is organized in a concept hierarchy containing 91 concepts. This makes the dataset complex enough to pose a real data integration challenge, but as we could also confirm in the study, small enough to be handled in a limited amount of time.

4.3.2 Target Schema

In order to be able to control the experiment and include different types of errors in the alignment, we manually built a target schema to which the records of the source dataset need to be translated. Building the schema, we followed established best practices for conceptual modeling.

The resulting schema consists of a 29 classes organized in a hierarchy (compare Figure 6) and 20 at-

[2]http://gaia.isti.cnr.it/~straccia/download/teaching/SI/2006/Autos.owl

tributes for describing data records.

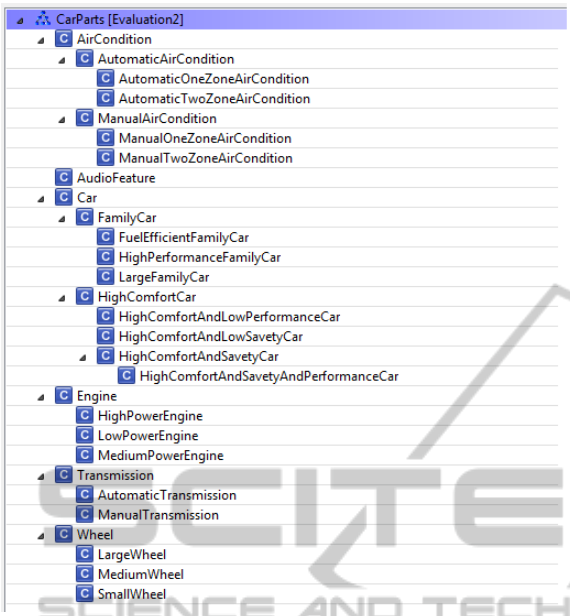

Figure 6: Class hierarchy of the manually created target schema.

4.3.3 Mapping Rules

We manually created a set of mapping rules between the two schemas and validated the correctness of the mappings with respect to correctly translated data. Based on this correct mapping set, we created two mapping sets each of which contain ten errors. We introduced errors with different level of complexity. The main task of the participants in the study was detecting as much of the produced errors as possible within the limited timeframe.

The simplest type of wrong mapping rules are rules connecting classes that are not identical. An example is the mapping rule `wheel` on `engine`. We assume that these kinds of errors are easy to spot even by inexperienced users. Out of ten errors each dataset contained four errors of such a kind.

The second type of errors was introduced by modifying the filter conditions associated with a correct mapping rule. An example would be the rule mapping `AirCondition` to `AutomaticOneZoneAirCondition`. The correct filter conditions for this mapping rule are `hasZoneNumber = 1` and `hasAutomatic = true`. In this case we modified the filter conditions to `hasZoneNumber = 2` and `hasAutomatic = false`. We assume that these kinds of errors are harder to identify by the user because it requires a detailed investigation of the mapping definition. Each of the two mapping sets contained six out of ten errors of this type.

4.3.4 Domain Information

Participants were provided with background knowledge about the domain of interest. The knowledge consisted of information about specific car types and car equipment. The respective information was provided in terms of images in combination with short descriptions. Figures 7 and 8 show examples of such information for cars and wheels.

Figure 7: Simulated background knowledge about cars.

Figure 8: Simulated background knowledge about wheels.

4.4 Participants

Twenty-two subjects participated in the user study. Six of the subjects were female and sixteen male. The average age of the subject was 27.8 year with the youngest subject being 21 and the oldest over 50. About half of the subjects (54% of the persons involved in the study) were students.

All participants were used to utilize complex software tools (average: 4.91). On average subjects had medium experience with using filter rules (average: 3.72), conceptual models (average: 3.13), and data integration (average: 2.72) in the past.

In all three cases, the answers ranged from 1 to 5 providing a good coverage of different levels of expertise. In particular, the variance of the subjects' experience using filter rules was 1.92, using conceptual models was 2.22, and using data integration was 1.79.

5 EXPERIMENTAL RESULTS

We analyzed the results of the study with respect to the quality of the results produced by the participants, the correlation of the results with the level of expertise and the perceived support by the system. Since all users perform both integration tasks, we can directly compare the users' performance on the respective integration tasks.

The detailed results are presented in the following subsections.

5.1 Quality of the Results

We measure the quality of the results for an individual participant by comparing the errors identified by the user with the set of errors previously introduced in the mapping set. We use well known quality measures from the area of information retrieval, more specifically precision, recall and F^2-measure. Before presenting the results, we briefly recall the definition of these measures.

Let TP denote the number of true positives, that is the number of errors that have correctly been identified by a subject, FP the number of false positives that is the number of mapping rules that have falsely been identified by the subject as an error and FN the number of false negatives, namely the number of errors that have not been found by the subject. Following this definition precision (P), recall (R) and F^2 measure (F^2) are defined as follows:

$$P = \frac{TP}{TP+FP} \tag{1}$$

$$R = \frac{TP}{TP+FN} \tag{2}$$

$$F^2 = \frac{(1+2^2)P \cdot R}{(2^2)P+R} = \frac{5P \cdot R}{4P+R} \tag{3}$$

Figure 9 compares precision, recall and F2 measure that have been achieved by the subjects on average for the two integration tasks.

The results show that there was no significant difference between the conventional tool and the MappingAssistant approach with respect to precision. Both values are close to 1.0 meaning that subjects almost never identify mappings as errors that are actually correct. We can see however, that there is a significant difference with respect to recall. Using conventional technology, the subjects were only able to find two thirds of the errors on average. In comparison to that, using our approach the yielded recall was much higher than utilizing conventional technology. In particular, the subjects on average were able to find

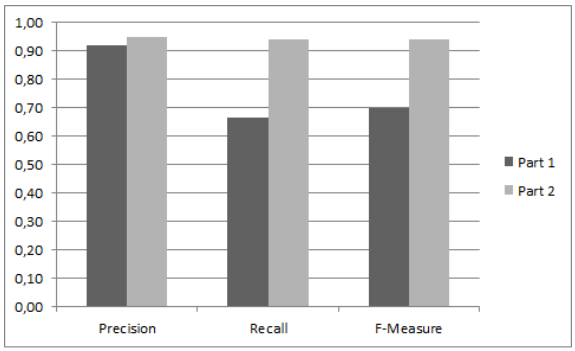

Figure 9: Average performance of subjects on the two integration tasks.

nine out of ten errors within the given time frame of ten minutes.

The advantage of the MappingAssistant approach with respect to identifying existing errors more efficiently becomes even more obvious when looking at Figure 10. It shows that 91% of the subjects found more errors using MappingAssistant than with the conventional tool. 9 % of subjects found the same number of errors and none of the subjects showed a better performance with the conventional technology.

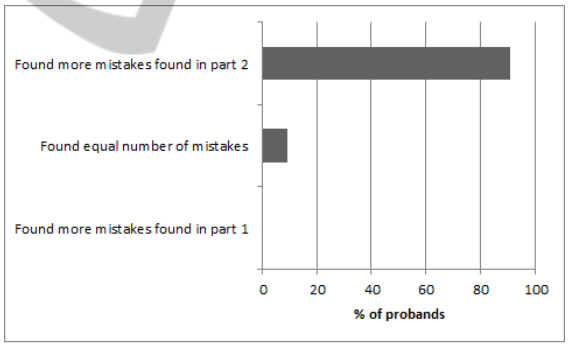

Figure 10: Comparison of performance on subject level.

The results clearly show that our method helps users to find more errors in a given period of time thereby confirming our hypothesis that the method increases efficiency and effectiveness of data integration.

5.2 Correlation with Level of Expertise

In order to test our hypothesis that our method in particularly supports users with limited technical knowledge, we compared the performance of participants with their previous knowledge in conceptual modeling and data integration. When comparing previous knowledge in conceptual modeling and data integra-

tion with the overall performance in task 1, we can see that there is indeed a relation between these two aspects (compare Figure 11).

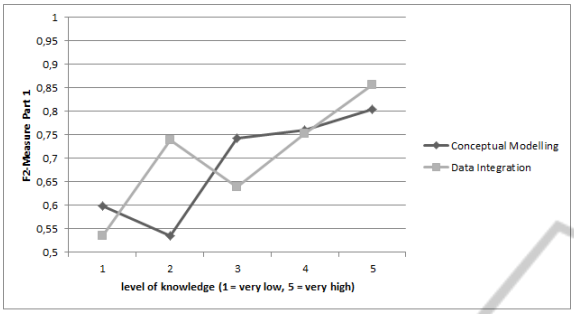

Figure 11: Relation between previous knowledge and performance in integration task 1.

We can conclude that for successfully performing data integration with a conventional tool a high degree of expertise in conceptual modeling and data integration is needed. While the trend is not that clear with respect to previous knowledge in conceptual modeling, the result is more conclusive with respect to previous knowledge in data integration.

Figure 12 shows the relation between the level of previous knowledge and the performance improvement achieved by using the MappingAssistant approach.

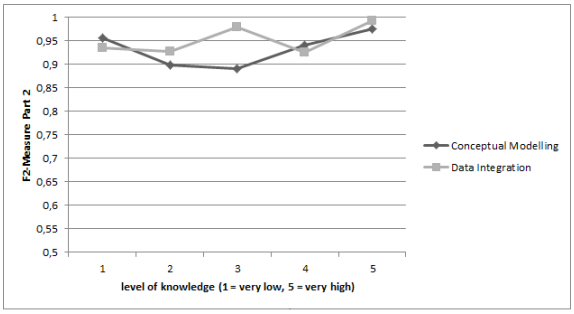

Figure 12: Relation between previous knowledge and performance in integration task 2.

With the MappingAssistant approach the subjects achieved high F^2 measure results. The results of the test persons were independent from the previous degree of expertise in conceptual modeling as well as data integration.

Although the results are not as clear as for the quality improvement, we can observe that the performance using the MappingAssistant is independent from previous knowledge. In summary, these results confirm our hypothesis that a user-centric interactive approach to data integration has a stronger positive effect for people with little technical knowledge compared to the conventional approach.

5.3 Qualitative Results

In addition to the objective results, we were also interested in how the different tasks were perceived by the users. In particular, we wanted to find out, how intuitive the user interface was and how the users judged the difficulty and the support by the systems. In the following we present the answers to several questions related to the mentioned aspects.

Figure 13 shows the average answer of the users with respect to task 1 according to the 1-5 scale where 1 means *complete disagreement* and 5 *complete agreement*.

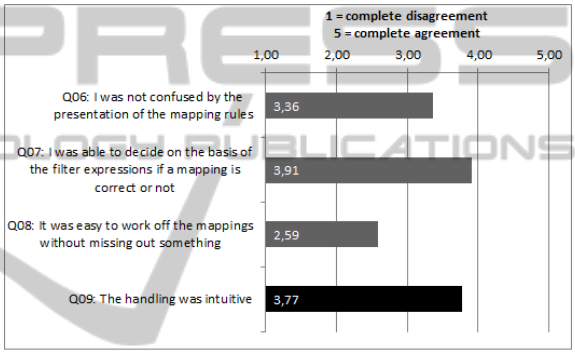

Figure 13: User feedback on task 1.

The results show that users found difficult to complete the task without missing mapping rules. With respect to the other questions about the difficulty of the task and the design of the user interface, the users were rather indifferent. Figure 14 shows the answers of users with respect to task 2. Here we see that in general there is a much higher agreement of users with questions concerning the benefits of the approach. In particular, users considered the MappingAssistant approach to be more intuitive than the traditional one (3.77 vs. 4.55 average score) supporting our hypothesis.

While the results are not directly comparable, we can still conclude that our assumption about the benefits of the MappingAssistant approach being more intuitive and easier to follow by the users is shared by the subjects of our study.

6 CONCLUSIONS

Data integration is a difficult task that typically requires substantial knowledge not only of the data to

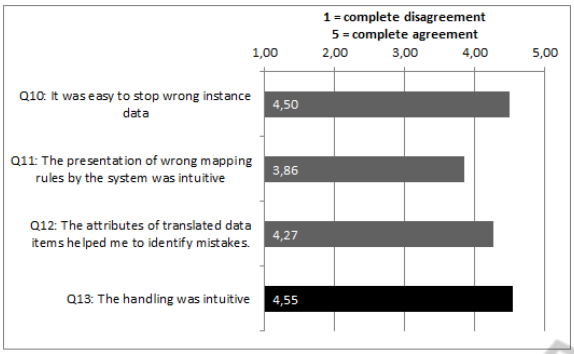

Figure 14: User feedback on task 2.

be integrated but also of data integration technologies. The goal of our research was to enable the people with less or no knowledge of these technologies to integrate their data. We presented a user-centric approach to data integration that is based on a cognitive support model, which has been implemented in the MappingAssistant data integration tool. We presented the results of a user study demonstrating that the approach empowers users to solve data integration problems more effectively and efficiently. In particular, we showed that users were able to find more errors in mapping rules in a given period of time. Further, we were able to show that while with conventional mapping technology a high level of expertise in mapping technology is required, the MappingAssistant approach significantly reduces the performance difference of experienced and inexperienced users. Finally, we could show that users considered our approach to be more intuitive.

We believe that the user-centric approach to data integration presented in this paper can have a real impact on the practice of data integration in enterprises. In particular, the approach can help expert users within an organization to retain more responsibility for data integration. While today, data integration tasks often either have to be delegated to the IT department or even be outsourced to specialized companies, our technology can enable users to perform non critical data integration tasks themselves. This can save time and money in enterprise data integration. Furthermore, it can create more options for on the fly data integration or mesh-ups that can provide useful information but are not needed on a regular basis.

In future work we will extend the presented diagnosis component of the MappingAssistant with an induction component. This induction component will provide suggestions for repairing the wrong mapping rule which has previously been found with the diagnosis component.

REFERENCES

Bellahsene, Z., Bonifati, A., and Rahm, E. (2011). *Schema Matching and Mapping*. Springer Publishing Company, Incorporated.

Cruz, I., Antonelli, F., and Stroe, C. (2009). Agreementmaker: efficient matching for large real-world schemas and ontologies. *Proceedings of the VLDB Endowment*, 2(2):1586–1589.

Euzenat, J. and Shvaiko, P. (2007). *Ontology matching*. Springer-Verlag New York Inc.

Euzenat, J. and Ferrara, A. and Hollink, L. and Isaac, A. and Joslyn, C. and Malaisé, V. and Meilicke, C. and Nikolov, A. and Pane, J. and Sabou, M. and others (2010). Results of the ontology alignment evaluation initiative 2010.

Falconer, S. (2009). Cognitive support for semi-automatic ontology mapping. *PhD Thesis, University of Victoria*.

Falconer, S. and Noy, N. (2011). Interactive techniques to support ontology matching. *Schema Matching and Mapping*, pages 29–51.

Falconer, S. and Storey, M. (2007). A cognitive support framework for ontology mapping. *The Semantic Web*, pages 114–127.

Fallahi, F., Noessner, J., Kiss, E., and Stuckenschmidt, H. (2011). Mappingassistant: Interactive conflict-resolution for data integration. *Poster at the 8th Extended Semantic Web Conference, ESWC*.

Gass, O. and Maedche, A. (2011). Enabling end-user-driven data interoperability - a design science research project. In *Proceedings of the 17th Americas Conference on Information Systems (AMCIS 2011), Detroit, USA*.

Granitzer, M., Sabol, V., Onn, K. W., Lukose, D., and Tochtermann, K. (2010). Ontology alignment - a survey with focus on visually supported semi-automatic techniques. *Future Internet*, 2(3):238–258.

Hair Jr, J., Anderson, R., Tatham, R., and Black, W. (1995). *Multivariate data analysis*. Prentice-Hall, Inc.

Halevy, A., Rajaraman, A., and Ordille, J. (2006). Data integration: The teenage years. In *Proceedings of the 32nd international conference on Very large data bases*, pages 9–16. VLDB Endowment.

Maier, A., Schnurr, H., and Sure, Y. (2003). Ontology-based information integration in the automotive industry. *The Semantic Web-ISWC 2003*, pages 897–912.

Noessner, J., Fallahi, F., Kiss, E., and Stuckenschmidt, H. (2011). Interactive data integration with mappingassistant. *Demo Paper at the 10th International Semantic Web Conference ISWC*.

Teevan, J., Jones, W., and Gwizdka, J. (2006). Personal information management. *Communications of the ACM*, 49(1):68–73.

Von Alan, R., March, S., Park, J., and Ram, S. (2004). Design science in information systems research. *Mis Quarterly*, 28(1):75–105.

Wohlin, C. (2000). *Experimentation in software engineering: an introduction*, volume 6. Springer.