FuSem – Exploring Different Semantics of Data Fusion

Jens Bleiholder
Hasso-Plattner-Institut, Potsdam
jens.bleiholder@hpi.uni-potsdam.de

Karsten Draba
Hasso-Plattner-Institut, Potsdam
karsten.draba@hpi.uni-potsdam.de

Felix Naumann
Hasso-Plattner-Institut, Potsdam
felix.naumann@hpi.uni-potsdam.de

ABSTRACT

Data fusion is the final step of a typical data integration process, after schematic conflicts have been overcome and after duplicates have been correctly identified. We present the relational data fusion system *FuSem*, which uses schema mappings and information about duplicates to decide *what* to fuse, i.e., which tuples to merge into one. The aspect emphasized by the demo is *how* to fuse the duplicates with *FuSem*. First, it offers several conflict resolution functions to handle data conflicts among duplicates. Furthermore, different fusion semantics proposed in the literature, such as MatchJoin or ConQuer, can be compared and visually explored. Optimized execution allows interactive access to the data and thus to explore the different data fusion procedures.

1. DATA FUSION

Integrated information systems enable users to query different heterogeneous data sources with a single query. Reformulating the query, sending it to the different sources and presenting the result to the user are tasks automatically performed by such a system. In order to generate a final result, the system must solve three basic problems [9]: 1. Identify and map semantically equivalent schema elements between sources (*schema matching & mapping*). 2. Identify and map same real world objects that are stored in same/different sources (*duplicate detection*). 3. Resolve potentially different representations of one and the same real-world object (data conflicts or uncertainties) using *data fusion* techniques, which are the focus of *FuSem*.

Research has suggested several techniques and semantics to handle data conflicts, once schema and object equivalences are established. In our demo we show how users can query dirty data sources and obtain a clean result by applying several alternative data fusion semantics. Because these semantics have different properties and outcomes, we provide the user with a tool to visually compare different results of fusion operations. Together with an interactive query interface, fast query execution, and an intuitive presentation of the results, the system allows easy exploration of fusion results, the underlying data sources, and applying and testing different data fusion operations to finally come up with the most suitable fusion result for the task at hand.

FuSem – short for *Fusion Semantics* or just *fuse them* – is a Java application based on parts and an enhancement of the HumMer data integration system [1]. It allows to access different data sources and pose SQL-like queries. Complementing the original wizard interface of the HumMer [1] system, users of *FuSem* can specify the result completely as SQL-like queries. It assumes that schema matching and duplicate detection has been performed upfront. Queries to the system are optimized, whenever possible. The system incorporates the semantics of several related studies (Merge, MatchJoin, consistent answers, etc. detailed in Section 2) and understands the proposed syntax of each.

The system's architecture allows easy addition of new semantics for data fusion. We use XXL [5], a framework for implementing query processing functionality. In order to extend *FuSem* and add a new fusion semantics, three things need to be provided: First, a mechanism to specify a query, either in the form of a GUI, in the form of a parser, or in the form of a rewriting into the syntax of one of the already existing semantics. Second, a translation mechanism that creates an XXL operator tree out of the query, and third, a routine (e.g., in the form of an XXL cursor) that computes results using the new semantics. If desired, one can also specify transformation rules and cost/cardinality estimators for the optimizer. Access to sources, visualization, interaction between components, and the presentation of results is already provided by the system.

Data fusion results are presented as visually enhanced tables. They graphically show contradictions and uncertainties in the original data and visualize how the different semantics deal with those contradictions. To quickly test several fusion techniques, the system generates a sample of the data for visualization.

2. FUSION SEMANTICS

FuSem is able to handle inconsistent information from multiple heterogeneous sources by using five different approaches:

- The sources can be queried using standard SQL. Thus, it is possible to "fuse" data by applying an (outer-) union or an (outer-) join. More complex SQL statements are also possible, offering some, but generally only limited data fusion capabilities.

VLDB '07, September 23-28, 2007, Vienna, Austria.

- The system can perform the Merge operator as proposed in [8], combining two data sources by a series of outer join and union operations, creating a result with uncertainties removed, but that still may contain contradictions.
- *FuSem* also implements the MatchJoin operator, which uses an outer-join to combine all possible attribute values for an object and chooses among them given a confidence parameter [10], possibly still not resolving all contradictions at hand.
- *FuSem* supports the consistent query answering approach in form of the rewritings given by the ConQuer system [6][1]. This approach allows only non-contradictory tuples to enter the final result.
- Finally, *FuSem* offers to resolve inconsistencies by grouping and aggregation by means of the FuseBy statement presented as an SQL extension in [2].

Some of the approaches introduced (e.g., MatchJoin and FuseBy) allow for parameterization and therefore allow the user to apply different variants of the specific semantics to the data. For example, one can vary the conflict resolution functions when using FuseBy. Differences in the results when applying different semantics arise mainly because of the way data conflicts are handled, e.g., the number of representations that are included in the result in case of multiple representations per real world object (e.g., none for ConQuer, two or more for Merge). Other differences arise when different data values are chosen if only one representation is included in the result (e.g., MatchJoin, FuseBy with conflict resolution functions). More details on conflict handling strategies and their differences can be found in [3].

Figure 1 shows the user interface of *FuSem*, where queries are issued to the sources under different fusion semantics – one window for each query, using one of the possible semantics. Thereafter, the interface can graphically display the query execution plan and of course the query result table itself. One option to compare results of different approaches is to align the corresponding result tables next to each other. More sophisticated comparison methods are discussed in the next section.

3. COMPARING FUSION RESULTS

Once the user has created several fused results using different techniques, the system is able to compare these. *FuSem* provides some numerical information when comparing fused results and in particular graphical access to the fused results.

3.1 Exploring Differences

We employ Venn-diagrams to show the overlap of object descriptions, similarities, and contradictions in the representations of fused results. To illustrate, regard the example in Table 1. For the four objects o_1 – o_4 we show the two attributes Description and Size as they would appear under the respective semantics.

Figure 2 shows three different visualizations of the data of Table 1 as presented by *FuSem*. Each visualizes a different aspect of data fusion; each is a more detailed view of the previous.

[1]We thank Ariel Fuxman for kindly providing the source code.

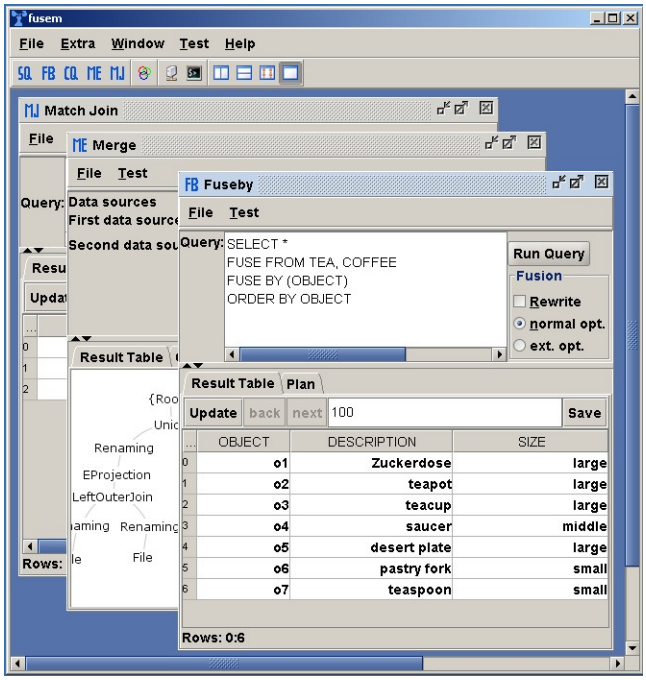


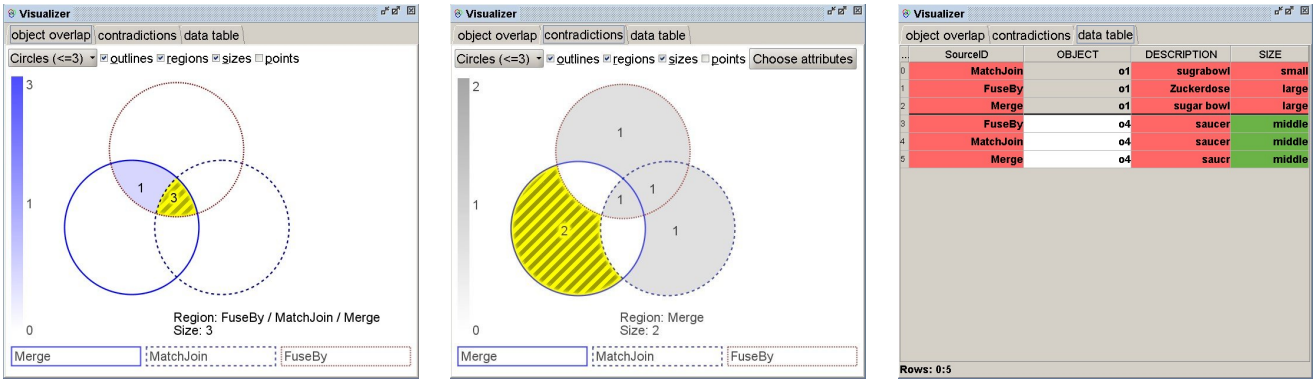
Figure 1: Screenshot of *FuSem*, which shows three of five different data fusion techniques along with their respective results and the plan view for one approach.

Visualizing object overlap: In Figure 2(a) we show the Venn-diagram of three regions, each corresponding to one of the three fusion semantics of the example – Merge, FuseBy, and MatchJoin. Each region contains the set of result tuples. The number of objects represented by them is shown in the corresponding region in the diagram. If one set contains object representations that also appear in another set, this is represented by overlapping regions. For instance, objects o_1, o_2, and o_4 are located in the highlighted sub-region towards the center of the diagram. All three objects are in the results of all three semantics. Thus they are placed in the overlap of all three regions.

By clicking on one of the sub-regions (e.g., the one in the middle, highlighted in Figure 2(a)) a second view details the actual objects of that region, and in particular, their tuple overlap.

Visualizing tuple overlap: This second view (Figure 2(b)) visualizes the degree of data conflicts among object representations. Again, each region in the Venn-diagram represents a fusion semantics. If two object representations from different semantics are equal (same Description and Size values) they appear in the corresponding overlap of their regions. For instance, object o_2 has the same Description and Size values in all three fusion semantics, which puts it into the overlapping sub-region of all three regions. Again, we show only the number of objects which are contained in a certain sub-region. Object o_1 on the other hand differs in all three semantics, which is the reason why it is placed into the three outer regions, which have no overlap.

Which attributes are considered in determining conflicts can be specified by the user. This ability allows rapid exploration of the degree of conflict in individual or certain



(a) Object overlap (b) Contradiction Count (c) Contradictory tuples

Figure 2: Visualization of the example fusion results from Table 1. We show (a) object overlap in fusion results, (b) contradictions, and (c) contradictory tuples.

Object	Semantics	Description	Size
o_1	Merge	sugar bowl	large
	FuseBy	Zuckerdose	large
	MatchJoin	sugrabowl	small
o_2	Merge	teapot	large
	FuseBy	teapot	large
	MatchJoin	teapot	large
o_3	Merge	teacup	large
	FuseBy	teacup	large
o_4	Merge	saucr	middle
	FuseBy	saucer	middle
	MatchJoin	saucer	middle
...	...	...	...
	...	...	...

Table 1: Example data showing object representations as produced using different fusion semantics.

combinations of attributes. The specific conflicting tuples with their data values can be explored by again selecting a certain sub-region.

Visualizing data conflicts: In the table view of Figure 2(c) we present all representations of the set of objects in the sub-region chosen in the previous view (e.g., the lower left one, highlighted in Figure 2(b)). Thus a user can determine the severity of the actual data conflicts, and if needed change the original queries and finally select the desired fusion semantics.

3.2 Visualization Variations

Exploring source conflicts: The visualization interface of *FuSem* can also be used to visualize the overlap and inconsistencies among different data sources: Instead of applying different fusion semantics and comparing them, we assume each data source to represent a "semantics" and compare those. Note that we must still assume a common key to identify objects.

Objects and colors: *FuSem* has the ability to display objects as points in the respective sub-regions. Figure 3(a) shows an example for six different fusion results using square-style Venn-diagrams. That way agreement or disagreement between different fusion semantics can easily be spotted by point clusters in the regions. As was already shown in previous figures, numbers of objects are not only represented as integers or as sets of points, but are also represented by coloring sub-regions (see Figure 3(b)).

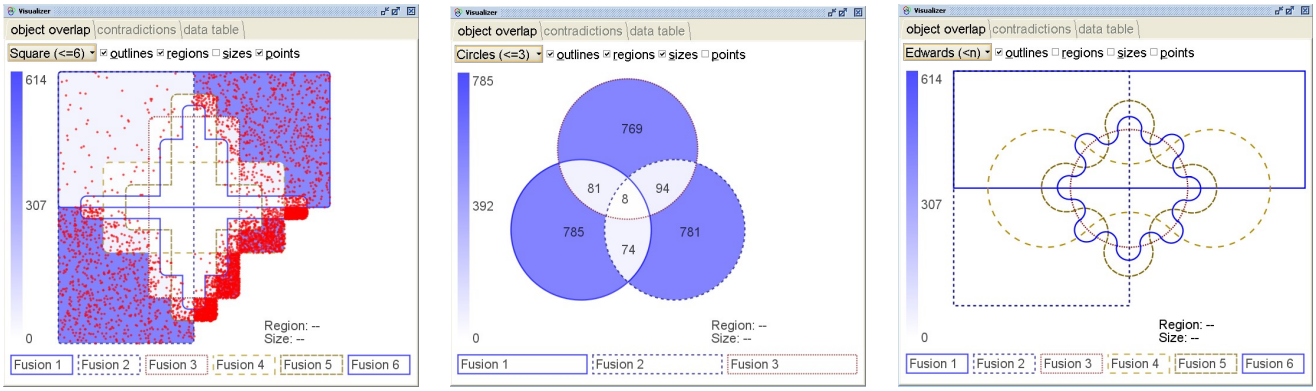
Venn-diagram types: The displayed technique of showing conflicting data is easily extended to higher numbers of different semantics (or data sources). Unfortunately, Venn-diagrams for more than five or six sets are difficult to cope with for the average user. However, in most relevant scenarios, the number of results to compare and therefore the number of sets to visualize will be small enough to not cause major problems. We have implemented three ways of drawing Venn-diagrams as presented in [7]. Figure 3(c) shows an Edwards-style Venn-diagram used for a visualization of six different fusion results. The visualization of area-proportional Venn-diagrams [4] is currently under development.

3.3 Statistical Data

Some basic statistics are presented to the user when comparing fused results:

- Agreement among object representations. We compute and present per sub-region the percentage of object representations that do not differ.
- Contradictions among object representations. On the other hand, the percentage of object representations that are contradictory are computed and shown per sub-region.
- Uncertainties among object representations. A special case of contradiction represents the contradiction of a value and a NULL value. We call this an uncertainty and the system also computes and presents the degree of uncertainties among object representations.

These statistics are presented to the user in form of a statistical summary and can be computed for different attributes or combinations of attributes thus emphasizing in which attributes the semantics differ most, or what percentage of objects are fused the same way using different semantics. The user only sees the parts of the statistics that coincide with the actual view. For instance, a low percentage of contradictions among different fusion results leads to



(a) Showing objects as points (b) Colorcoding counts (c) Comparing 6 fusion results

Figure 3: Example visualizations showing (a) square-style, (b) circle-style, and (c) Edwards-style Venn-diagrams.

a high similarity among the different semantics, inviting the user to purposefully explore only the small number of object descriptions where the fusion semantics differ in their results, or finally choose the fusion technique which is computationally cheaper while still creating the same (desired) result.

4. DEMO SCENARIOS

To present the five semantics of data fusion, and especially to show their similarities and differences we regard two scenarios and user tasks: A first scenario is from the customer-relationship-management domain (CRM) and demonstrates the fusion of customer and order data from several sources, using artificially generated data from the TPC-H benchmark and artificially introduced data conflicts.

The second scenario shows the fusion of different search engine results. Here, the search engines Google, MSN, and Yahoo are wrapped and brought into a relational format, each query resulting in a table with defined schema. A normalized URL is used to identify web pages (duplicates in the search results). Data conflicts, such as different sizes, titles, and other content, are handled during fusion.

In both scenarios qualitative aspects (how well do the fusion techniques work and how do they compare?) as well as quantitative aspects (how do the techniques scale?) play a role and are considered.

5. CONCLUSIONS

With *FuSem* we present an intuitive, interactive system to apply different data fusion semantics to data sources and visually explore their results, their similarities, and their differences. Thus, the system helps users to better understand their data and helps them to appropriately integrate different conflicting sources and finally come up with the most suitable fusion result for the task at hand.

Acknowledgment. This research was supported in part by the German Research Society (DFG grant no. NA 432).

6. REFERENCES

[1] A. Bilke, J. Bleiholder, C. Böhm, K. Draba, F. Naumann, and M. Weis. Automatic data fusion with HumMer (Demo). In *Proc. of VLDB*, pages 1251–1254, 2005.

[2] J. Bleiholder and F. Naumann. Declarative data fusion - syntax, semantics, and implementation. In *Proc. of ADBIS*, pages 58–73, 2005.

[3] J. Bleiholder and F. Naumann. Conflict handling strategies in an integrated information system. In *Proc. of IIWeb Workshop*, 2006.

[4] S. Chow and F. Ruskey. Drawing area-proportional Venn and Euler diagrams. In G. Liotta, editor, *Graph Drawing, Perugia, 2003*, pages pp. 466–477. Springer, 2004.

[5] J. V. den Bercken, B. Blohsfeld, J.-P. Dittrich, J. Krämer, T. Schäfer, M. Schneider, and B. Seeger. XXL - a library approach to supporting efficient implementations of advanced database queries. In *Proc. of VLDB*, pages 39–48, 2001.

[6] A. Fuxman, E. Fazli, and R. Miller. ConQuer: efficient management of inconsistent databases. In *Proc. of SIGMOD*, pages 155–166, 2005.

[7] A. Glassner. Venn and now. *IEEE Comput. Graph. Appl.*, 23(4):82–95, 2003.

[8] S. Greco, L. Pontieri, and E. Zumpano. Integrating and managing conflicting data. In *Revised Papers from the 4th International Andrei Ershov Memorial Conference on Perspectives of System Informatics*, pages 349–362, 2001.

[9] F. Naumann, A. Bilke, J. Bleiholder, and M. Weis. Data fusion in three steps: Resolving schema, tuple, and value inconsistencies. *IEEE Data Eng. Bull.*, 29(2):21–31, 2006.

[10] L. L. Yan and M. T. Özsu. Conflict tolerant queries in AURORA. In *Proc. of CoopIS*, page 279, 1999.