NEW CHALLENGES IN DATA INTEGRATION: LARGE SCALE AUTOMATIC SCHEMA MATCHING

[1*]Kamal Kant
Department of Computer Science & Engineering,
Madan Mohan Malviya University of Technology,
Gorakhpur, (UP) India.
Email: kk.kamal2525@yahoo.com

[2,3,4]A.K.Sharma, Kumar Ashis, Sanjay Kumar
Department of Computer Science & Engineering,
Madan Mohan Malviya University of Technology,
Gorakhpur, (UP) India.
Email: akscse@rediffmail.com

ABSTRACT: Today schema matching is a basic task in almost every data intensive distributed application, like enterprise information integration, collaborating web services, web catalogue integration and schema based P2P database systems. There has been a plethora of algorithms and techniques researched in schema matching and integration for data interoperability. Many surveys have been presented in the past to summarize this research. The requirement for extending the previous surveys has been created because of the mushrooming of the dynamic nature of these data intensive applications. Indeed, evolving large scale distributed information systems are further pushing the schema matching research to utilize the processing power not available in the past and directly increasing the industry investment proportion in the matching domain. This article reviews the latest application domains in which schema matching is being utilized. The paper gives a detailed insight about the desiderata for schema matching and integration in the large scale scenarios. Another panorama which is covered by this survey is the shift from manual to automatic schema matching. Finally the paper presents the state of the art in large scale schema matching, classifying the tools and prototypes according to their input, output and execution strategies and algorithms.

Keywords: Schema integration, data integration, Mappings, Schema Merging schema evolution; large scale.

1. INTRODUCTION

There exists an unending list of digital devices cooperating together to solve problems at individual level, personal or professional, and organisational level. The collaboration between these devices eventuates in better performance and results. Every day a new gadget hits the market, creating a ripple-effect in its surrounding operating environment. For the database community, it is an emergence of new form of data or information, which has to be utilised in the most efficient and effective manner. The ability to exchange and use of data/information between different devices (physical or logical), is the basic activity in any type of system, usually referred to as data interoperability.

Previous work on schema matching was developed in the context of schema translation and integration (Bernstein, Melnik, Petropoulos, & Quix, 2004; Do & Rahm, 2007; A. Halevy, Ives, Suciu, & Tatarinov, 2003), knowledge representation (Giunchiglia, Shvaiko, & Yatskevich, 2004; Shvaiko & Euzenat, 2005), machine learning, and information retrieval (Doan, Madhavan, Dhamankar, Domingos, & Halevy, 2003). All these approaches aimed to provide a good quality matching but require significant human intervention (Bernstein et al., 2004; Doan et al., 2003; Do & Rahm, 2007; Giunchiglia et al., 2004; A. Halevy et al., 2003; Lu, Wang, & Wang, 2005; Madhavan, Bernstein, & Rahm, 2001). However, they missed to consider the performance aspect, which is equally important in large scale scenario (large schema or a large number of schema to be matched).

By definition, schema matching is the task of discovering correspondences between

semantically similar elements of two schemas or ontologies (Do, Melnik, & Rahm, 2002; Madhavan et al., 2001; Milo & Zohar, 1998). Basic syntax based match definition has been discussed in the survey by Rahm and Bernstein (Rahm & Bernstein, 2001), extended by Shvaiko and Euzenat in (Shvaiko & Euzenat, 2005) with respect to semantic aspect. In this article, we discuss a new dimension of schema match, which focus on the requirements of automatic large scale schema matching and integration, also incorporating the previous ideas of mappings. We highlight the structural aspect of schema and its credibility for extraction of data semantics.

The requirement for enhancing the previous works of matching definition has been created because of the evolving large scale distributed information integration applications, which are also directly increasing the industry investment proportion (Davis, 2006)1 in the matching domain. The schema matching task of these applications which need to be automated are also discussed in length in this paper. Another aspect of this survey is the presentation of the schema matching classification from the perspective of latest strategies and algorithms in the field of schema based information retrieval and management.

II. New Application Domains for Data Interoperability

Schema matching research has its roots in schema integration applications in distributed database systems. The task is to produce a global schema from independently constructed schemas**.** The requirements for such integration have been presented in (Batini et al., 1986; Spaccapietra, Parent, & Dupont**, 1992).** The research highlights the issues in schema integration of relational schemas, the integrity of integrated schema and different possible techniques to integrate schemas (binary or n-ary). Data Warehousing, Message Translation (Rahm & Bernstein, 2001), E-commerce, B2B, B2C (Shvaiko & Euzenat, 2005) applications are examples of implementation of this research.

A. Web Services Discovery and Integration

The term that has been and will be repeatedly used throughout this dissertation is the term schema. By schema we mean any kind of structured or semi-structured representation of data. For example, Figure 1.1(a) illustrates schemas S^{er}_1 and S^{er}_2. Schema S^{er}_1 represents data about academic paper publications and schema S^{er}_2 represents data about academic text book publications. Now, schema integration is the activity of creating a single, unified representation of the schemas of multiple data sources, so that these data sources can be accessed transparently. The ultimate goal of schema integration is to provide interoperability between the data sources and make the retrieval of information and knowledge more efficient. The result of schema integration is an integrated schema and view definitions between the input schemas and the integrated schema.

B. Data Mashups in Enterprise Information Integration

Data Mashups is the most recent buzz word in the Enterprise Information Integration (EII) domain. Its definition can be: making new knowledge by joining available information. Web mashups are emerging at a rapid pace. *Programmable.com* provides a list of such mashups. A typical web mashup joins information from related web sites. For example a mashup website about cars can get quotes about a certain car from quotes websites, pictures and reviews from cars forums along with video footage from some social network like *youtube.com*. Thus the information resources can range from a simple database table to complex multimedia presentation i.e., the search can be on any structured or unstructured.

Thus the core concept in mashups is to extract some new necessary knowledge. From all these sources existing in different formats. This is a new challenging issue in information extraction and integration. The research aim is to provide light and fast protocols which can work through different meta models and. types of documents (A. Y. Halevy et al., 2006). At the enterprise level, the mashup idea helps in building quick situational applications, for some transient need in the enterprise, complementing the more

robust and scalable integration technologies that the enterprises invest in.

C. Schema based P2P Database Systems

One of the latest emerging research fields in databases over the web is *P2P Databases* (A. Halevy et al., 2003). There have been numerous successful P2P systems delivered in the last couple of years. Traditionally, the P2P systems have been simple file sharing systems which can self-tune, depending upon the arrival and departure of contributing peers. Industrial-strength file sharing P2P systems, like Kazaa and bitTorrent, allow the peer autonomy of participation but they still restrict the design autonomy of how to describe the data. Secondly, sharing of data objects described by one P2P system are not available in another P2P setup. Today, the P2P technology has transformed into sharing of any kind of data, whether it is semi structured XML data or continuous multimedia streaming (Meddour, Mushtaq, & Ahmed, 2006). The next generations of data sources are going to be totally independent of each other, i.e., they will have the design autonomy, utilizing their own terminologies for their data structuring, with capabilities to interact with others. For querying these data sources some matching method will be required to broker between their structures, giving rise to the new generation of application research of schema based P2P data sharing systems (Loser, Siberski, Sintek, & Nejdl, 2003).

C. Querying over the Web

Query processing has two intrinsic problems; understanding the query and then finding the results for it. The web contains vast heterogeneous collections of structured, semi-structured and unstructured data, posing a big challenge for searching over it. Deep Web (B. He et al., 2004) scenarios highlight this aspect. Firstly, the heterogeneity problem allows the same domain to be modeled using different schemas. As we have discussed in the example for our motivation. Secondly, it is very difficult to define the boundary of a domain. For example, traveling and lodging are inter-linked for tourist information web sites. Continuous addition of new content further complicates the problem for searching and integrating the results.

III. Schema Matching Techniques

This section gives an overview of the basic techniques used in the schema matching and integration research. Schema comprises of some basic entities called elements. The compositions of elements within the schema follow rules outlined by a data model.

A. Element Level

Schema matching is a complex problem, which starts by discovering similarities between individual schema elements. Every element, disregarding the level of granularity, is considered alone for a match. The techniques used, basically rely on the element's name and associated description, using basic **string matching** approaches adapted from the information retrieval domain (Duchateau, Bellahsene, & Roche, 2007). These approaches include string prefix, suffix comparisons, soundx similarities and more sophisticated algorithms based on string distance. There is a large list of these algorithms with various variations researched over time. he mainly talked about approaches are the n-gram and the edit distance3. For example Google use n-gram for statistical machine translation, speech recognition, spelling correction, information extraction and other applications. **Linguistic techniques** are based on the tokenisation, lemmatisation and elimination. The idea is to extract basic sense of the word used in the string. And then find its contextual meaning (Bohannon, Elnahrawy, Fan, & Flaster, 2006; Duchateau et al., 2007) i.e., meaning extraction according to the elements around it. These techniques have been adopted from linguistic morphological analysis domain. The algorithms are further enriched to provide synonym, hypernym, hyponym similarities by using external oracles, dictionaries, thesauri like WordNet (Gangemi, Guarino, Masolo, & Oltramari, 2003), domain specific ontologies or upper level ontologies (Niles & Pease, 2003).

B. Structure Level

Structure level matching is referred as matching a combination of elements from one schema to another schema (Rahm & Bernstein, 2001). The algorithms developed are based on graph matching research. It can also utilize external oracles like known patterns (Embley, Xu, & Ding, 2004), ontologies (Doan et al., 2003) or corpus of structures (Madhavan et al., 2005) to recognize the similarity. It also helps in solving n:m complex match problem. Today, almost every schema matching implementation uses some form of **graph structures** for internal representation of schemas. Graph matching is a combinatorial problem with exponential complexity. Researchers use directed acyclic graphs or trees to represent schemas, ontologies or taxonomies, to reduce the complexity aspect of the problem. In generic schema matching tools (which can take as input different data model schemas) the graph structures are flexible enough to support the possible input schema elements and perform mapping. Nearly all schema match research projects based on graphs use the notion of neighborhood affinity to compute the similarity match value for individual elements. This aspect has been presented in Similarity Flooding algorithm (Melnik, Garcia-Molina, & Rahm, 2002). In large scale scenarios, structure level matching techniques help in enhancing the performance of the match implementations, by using neighborhood search algorithms (Ehrig & Staab, 2004). In literature holistic (B. He et al., 2004) or level-wise algorithms (children-parent relationships) (Madhavan et al., 2001; Do & Rahm, 2007) have been used to determine the correspondences among two schemas.

C. Use of Data Instances and Machine Learning

Data instance in schema matching is used in two ways. First, if the schema information is very limited or not available, instance data is used to create a representation of the data (Bex, Neven, & Vansummeren, 2007). For example from any XML document, a basic tree hierarchy of elements can be extracted. Even, if the schema is available, data instances can augment the schema matching by giving more insight about the schema element semantics (Hernedez et al., 2002). For example city names encountered in data instances (found in a general list of city names) can infer that the field is a component of address field.

IV. Match Strategies

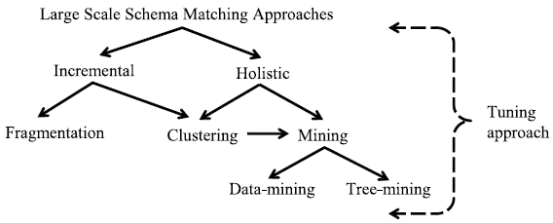

Fig.1: Taxonomy for Large Scale Schema Matching and Integration Strategies

Different schema match research projects have shown that single match algorithm is not enough to have a quality match. It is necessary to employ a range of algorithms, applied in a sequence or parallel, optimized for the application domain. Researchers have followed different strategies depending on application domain or researcher's objectives. The strategy is basically governed by the input and output requirements of the match tool.

A. Schema Fragmentation Approach

In the domain of semi-structured data, more and more schemas are being defined in XML, a standard language adopted by W3C. It is being widely used in E-business solutions and other data sharing applications over the web. Over time, emergences of distributed schemas and namespaces concepts have introduced more complexity to the matching problem. Research work in (Do & Rahm, 2007) demonstrates how these emergent problems can be tackled. The authors propose the idea of fragmentation of schemas for matching purposes. The approach first creates a single complete schema, including the instances for the distributed elements or namespaces used in the schema. In second step the large schema instance is broken down into logical fragments which are basically manageable small tree structures. The tool COMA++ (Aumueller et al., 2005) is used to compare each fragment from source schema to each fragment of target schema for

correspondences, with the help of GUI and human input. The approach decomposes a large schema matching problem into several smaller ones and reuses previous match results at the level of schema fragment. The authors have reported satisfactory results.

B. Clustering Approach

Clustering refers to the grouping of items into clusters such that items in one cluster are more similar to one another (high affinity) and those in separate clusters are less similar to one another (low affinity). The level of similarity can vary from application or technique which is using clustering approach. Since the schema matching problem is a combinatorial problem with an exponential complexity, clustering works as an intermediate technique and improves the efficiency of the large scale schema matching. In schema matching and integration, clustering can be considered at element level or schema level. *Element Level* clustering can be applied on a single schema or holistically on the given set of schemas. The authors of (Smiljanic et al., 2006) give a generic approach using the element level clustering method to detect element clusters in schema repository which are probably similar to a given personal source schema. Personal schema is then fully compared to detected list of clusters. So, rather comparing and applying all match algorithms on all schema elements in the repository, only a subset of elements are considered.

C. Data Mining Approach

Data Mining is the technique for finding similar patterns in large data sets. Very recently, it has been used as schema matching method. Work in (B. He et al., 2004; Su, Wang, & Lochovsky, 2006) highlight this method for matching and integrating deep web schema interfaces. (B. He et al., 2004) uses a positive correlational algorithm based on heuristics of schema attributes. Whereas (Su et al., 2006) applies negative correlational method to match and integrate schemas.

Tree mining approach is a variation of data mining, in which data is considered to possess a hierarchical structure. It shows more affinity to XML schemas, which are intrinsically tree structures. (Saleem et al., 2008) demonstrates a method which combines the element clustering and a tree mining method. The work provides a time performance oriented solution for integrating large set of schema trees, resulting in an integrated schema along with mappings from source to the mediated schema.

D. Strategies for Enhancing Match Results

There have been a lot of work on schema matching but proofs of exact results in the semantic world have been hard to achieve. In most of the research the quality of results has been said to be approximate (Rahm & Bernstein, 2001; Noy, Doan, & Halevy, 2005; Shvaiko & Euzenat, 2005). As a result of these observations new avenues of research opened up for finding ways to achieve the maximum correctness in schema matching. Following are the approaches under active research.

***Pre-Match Strategies*:** Pre-match methods typically deal with the matching tool's execution strategies, called *tuning match strategies*. These approaches try to enhance the performance of current schema matching tools which have the ability to rearrange the hybrid or composite execution of their match algorithms. Defining external oracles, the criteria for their use and adjustment of parametric values, like thresholds, for different algorithms is also part of pre-match. The work in (Y. Lee et al., 2007) provides a framework capitalizing on instance based machine learning. The authors describe, how the use of synthetic data sets can equip the matching tool with the ability to perform well, when applied to a similar real scenario. The tuning module execution is totally separate from the actual tool working.

***Post-Match Strategies*:** These strategies are concerned with improving the already obtained results from a schema matching tool. OMEN (Mitra et al., 2005) Ontology Mapping Enhancer, provides a probabilistic framework to improve the existing ontology mapping tools using a bayesian network. Is uses pre-defined meta-rules which are related to the ontology structure and the meanings of relations in the ontologies. It works on the probability that if one know a mapping between two concepts from the source

ontologies (i.e., they match), one can use the mapping to infer mappings between related concepts i.e., match nodes that are neighbors of already matched nodes in the two ontologies.

V. Overview of Large Scale Schema Matching Tools

The previous surveys (Rahm & Bernstein, 2001; Shvaiko & Euzenat, 2005; Yatskevich, 2003) incorporate solutions from schema level (metadata), as well as instance level (data) research, including both database and artificial intelligence domains. Most of the methods discussed in these surveys compare two schemas and work out quality matching for the elements from source schema to target schema.

A. Tools: Matching Two Large Schemas

COMA++ (Aumueller et al., 2005) is a generic, composite matcher with very effective match results. It can process the relational, XML, RDF schemas as well as OWL ontologies. Internally it converts the input schemas as graphs for structural matching and stores all the information in MYSQL as relational data. At present it uses 17 element/structure level matchers which can be selected and sequenced according to user's requirements.

PROTOPLASM (Bernstein et al., 2004) target is to provide a flexible and a customizable infrastructure for combining different match algorithms. Currently CUPID (Madhavan et al., 2001) implementation and Similarity Flooding (SF) (Melnik et al., 2002) algorithms are being used as the basematchers. A graphical interface for it has been proposed and demonstrated by the name of BizTalk Mapper (Bernstein et al., 2006). It is based on the HCI research presented in (George G. Robertson, 2005) and is very heavily dependent on microsoft technologies.

CLIO (Hernedez et al., 2002) has been developed at IBM. It is a complete schema mapping and management system. It has a comprehensive GUI and provides matching for XML and SQL schemas (Object Relational databases converted into relational with the help of a wrapper function). It uses a hybrid approach, combining approximate string matcher for element names and Naïve Bayes learning algorithm for exploiting instance data.

GLUE (Doan et al., 2003) is the extended version of *LSD* (Doan et al., 2001), which finds ontology/ taxonomy mapping using machine learning techniques. The system is input with set of data instances along with the source and target taxonomies. Glue classifies and associates the classes of instances from source to target taxonomies and vice versa.

LSD has been further utilized in *Corpus-based Matching* (Madhavan et al., 2005), which creates a corpus of existing schema and their matches. In this work, input schemas are first compared to schemas in the corpus before they are compared to each other. Another extension based on LSD is *IMAP* (Dhamankar et al., 2004). Here the work utilize LSD to find 1:1 and n:m mapping among relational schemas.

B. Tools: Matching and Integrating Large Set of Schemas

MOMIS (Beneventano, Bergamaschi, Guerra, & Vincini, 2001) is a heterogeneous database mediator. One of its components ARTEMIS is the schema integration tool which employs schema matching to integrate multiple source schemas into a virtual global schema for mediation purposes. The tool operates on hybrid relational-OO model. It first calculates elements similarity based on name and data type, thus acquiring all possible target elements.

Wise-Integrator (H. He, Meng, Yu, & Wu, 2004) is a schema integration tool. It uses schema matching to find correspondences among web search forms so that they can be unified under an integrated interface. First a local interface is selected and then incrementally each input form is compared against it. The attributes without a match candidate in the local interface, are added to it. Wise-Integrator employs several algorithms to compute attribute similarity. Namely exact and approximate string matching, along with dictionary lookup for semantic name similarity.

DCM framework (Dual Correlation Mining) (B. He et al., 2004) objective is similar to Wise-Integrator. It focuses on the problem of

obtaining an integrated interface for a set of web search forms holistically. The authors observe that the aggregate vocabulary of schemas in a (restricted) domain, such as book, tends to converge at a small number of unique concepts, like author, subject, title, and ISBN; although different interfaces may use different names for the same concept. The research proposes a statistical approach, extracted from data mining domain, based on the assumptions: independence of elements, non-overlapping semantics, uniqueness within an interface, and the same semantics for the same names.

PSM (Parallel Schema Matching)(Su et al., 2006), is another implementation of holistic schema matching, for a given set of web query interface schemas. The objectives are similar to DCM algorithm, but PSM improves on DCM on two things; first DCM negative correlation computation between two elements to identify synonyms may give high score for rare elements but PSM does not. And secondly the time complexity of DCM is exponential with respect to the number of elements whereas for PSM it is polynomial.

ONTOBUILDER (Roitman & Gal, 2006) is a generic multipurpose ontology tool, which can be used for authoring, and matching RDF based ontologies. Its interface also supports the process of matching web search forms for generating an integrated form. OntoBuilder generates dictionary of terms by extracting labels and field names from web forms, and then it recognizes unique relationships among terms, and utilize them in its matching algorithms.

VI. Summarizing the Tools

Table1: Schema Matching Tools and Prototypes Comparison – General

Tool	GUI	Approach	Card.	Ext Orc	Internal Rep	Research Domain
BELLFLOWER	No	Hybrid	1:1	-	Directed Graph	Schema Matching
CLIO	Yes	Hybrid	1:1	-	Rel. Model, Directed Graph	Schema Matching, Mapping Evolution
COMA++	Yes	Composite	1:1	Dom Syn, Abr Thesuri	Directed Graph	Schema Matching and Merging
DCM	No	Hybrid	n:m	-	-	Schema Integration
GLUE	No	Composite	n:m	-	Attribute based	Data Integration
MOMIS	Yes	Hybrid	n:m	Thesuri	Directed Graph	Schema Integration
ONTO BUILDER	Yes	Hybrid	1:1, 1:n	-	Graph	Create/Match Ontologies
PORSCHE	No	Hybrid	1:1,1:n	Dom Syn, Abr Thesuri	Tree	Schema Integration and Mediation
PROTOPLASM	Yes	Hybrid	1:1	Wordnet	Graph	Schema Matching
PSM	No	Hybrid	n:m	-	-	Schema Integration
QOM	No	Hybrid	1:1	Dom. Thesuri	Tree	Ontology Alignment
SCIA	Yes	Hybrid	n:m	Thesuri	Tree, Graph	Data Integration
WISE INTEGRATOR	Yes	Hybrid	1:1	General Thesuri	Attribute based	Web Search form Integration

It appears that the most prototypes aim to provide good quality matchings, with lack in time performance. Today, the application domains like the genomic or e-business, deal with large schema. Therefore the matching tool should also provide good performance and if possible automatic mapping generation. In future, matching systems should try to find a tradeoff between quality and performance.

Table2: Schema Matching Tools and Prototypes Comparison - Strategy based

Tool	Input	Output	Match Algorithms (Level wise)			
			Element			Stucture/(Data Ins.)
			Str.	*Ling.*	*Const.*	
BELLFLOWER	XSD	Schema Matches	Yes	-	-	K-means data mining
CLIO	SQL,XSD	Mappings (Query)	Yes	-	Yes	(Naive Byes Learner)
COMA++	XSD,XDR, RDF,OWL	Mappings, Merged Schema	Yes	Yes	Yes	Path: biased to leaf nodes
DCM	Web Query Interface	Mappings between all input schemas	Yes	-	Yes	Correlational Mining
GLUE	DTD,SQL, Taxonomy	Mappings, IMap functions	Yes	-	Yes	(Whirl/Bayesian Learners)
MOMIS	Rel,OO data model	Global View	Yes	Yes	Yes	Schema Clustering, Neighborhood Affinity
ONTO BUILDER	RDF	Mediated Ontology	Yes	Yes	-	Elements Sequencing
PORSCHE	XSD Instance	Mediated Schema	-	Yes	-	Elements Clust, Tree Mining
PROTOPLASM	XDR, SQL,RDF	Mappings	Yes	Yes	Yes	Path (Parent,Child,Grand Child), Iterative Fix Point Computation
PSM	Web Query Interface	Mappings between all input schemas	Yes	-	Yes	Correlational Mining
QOM	RDF(S)	Mappings	Yes	-	Yes	Neighborhood Affinity, Taxonomic Structures
SCIA	Rel,DTD, XSD,OWL	Mappings (Query)	Yes	Yes	Yes	Iterative Fix Point Computation, Path
WISE INTEGRATOR	Web Query Interface	Integrated Schema	Yes	Yes	Yes	Clustering

VII. Conclusion and Perspective

In this paper we provide a broad overview of the current state of the art of schema matching, in

the large scale schema integration and mediation for data interoperability. The paper also tries to provide an insight on current emergent technologies driving the match research.

We conclude our discussion by enumerating some explicit future research concerns in the field of schema matching and integration.

Maintenance of mappings with schema evolution.

- Visualization of mappings in multi-schema (more than 2) integration.
- Development of correctness/completeness metrics and benchmark tools for evaluating schema matching systems.
- Self-tuning of the matching tools, providing a balance between the quality and the performance aspects.

References

1. An, Y., Borgida, A., Miller, R. J., & Mylopoulos, J. (2007). A semantic approach to discovering schema mapping expressions. In *Intl. conf. on data engineering.*

2. Aumueller, D., Do, H. H., Massmann, S., & Rahm, E. (2005). Schema and ontology matching with coma++. In *Acm sigmod* (p. 906-908).

3. Bachlechner, D., Siorpaes, K., Fensel, D., & Toma, I. (2006). *Web service discovery - a reality check.* (Tech. Rep.). Digital Enterprise Research Institute (DERI).

4. Batini, C., Lenzerini, M., & Navathe, S. B. (1986). A comparitive analysis of methodologies for database schema integration. *ACM Computing Surveys*, *18* (4), 323-364.

5. Beneventano, D., Bergamaschi, S., Guerra, F., & Vincini, M. (2001). The momis approach to information integration. In *Iceis* (p. 194-198).

6. Benkley, S., Fandozzi, J., Housman, E., & Woodhouse, G. (1995). Data element tool-based analysis (delta). In *Mtr.*

7. Bernstein, P. A., Melnik, S., & Churchill, J. E. (2006). Incremental schema matching. In *Vldb.*

8. Bernstein, P. A., Melnik, S., Petropoulos, M., & Quix, C. (2004). Industrial-strength schema matching. *ACM SIGMOD Record*, *33* (4), 38-43.

9. Besana, P., Robertson, D., & Rovatsos, M. (2005). Exploiting interaction contexts in p2p ontology mapping. In *P2pkm.*

10. Bex, G. J., Neven, F., & Vansummeren, S. (2007). Inferring xml schema definitions from xml data.

11. Bilke,A.,&Naumann,F.(2005).Schema matching using duplicates.In *Intl. conf. on data engineering.*

12. Bohannon,P.,Elnahrawy,E.,Fan,W.,& Flaster, .(2006). Putting context into schema matching. In Vldb.

13. Dalamagasa,T., hengb, T., Winkele,K.J.,&Sellisa, T . (2006).A methodology for clustering xml document by structure. *Information System (Elseveir),*31,187228.

14. Davis, M. (2006). Semantic wave 2006-A guide to billion dollar markets-Keynote Address. In Stc.

15. Do, H.H., Melnik, s., &Rahm,E. (2002). Comparision of schema matching evaluation. In Web, *web-services, and database systems workshop.*