An Approach for the Extraction of Information from Heterogeneous Sources of Textual Data*

Sonia Bergamaschi
DSI - Università di Modena - CSITE-CNR
sonia@dsi.unimo.it,sbergamaschi@deis.unibo.it

Claudio Sartori
DEIS - Università di Bologna - CSITE-CNR
csartori@deis.unibo.it

Abstract

Extracting informations from multiple sources of textual data and integrating them in order to provide *information* is a challenging research topic in the database area. This paper presents a *Description Logics* approach to provide solutions both for data integration and data querying. The approach includes: a common description of sources, compliant with a subset of ODMG93; Description Logics techniques to optimize information extraction and to implement *mediators*(i.e. components which integrate and refine the data coming from the different sources).

1 Introduction

Extraction of heterogeneous textual data is, at present, heavily investigated in the database area, involving many research topics and application areas: decision support systems (DSS), integration of heterogeneous databases, data warehouse. Decision makers need informations from multiple heterogeneous sources (including databases, file systems, knowledge bases, digital libraries, information retrieval systems, and electronic mail systems), but are usually unable to get and fuse them in a timely fashion due to the difficulties of accessing the different systems and to consistently integrate them. Significant contributions about the integration of well-structured conventional databases exist (e.g. [A$^+$91]). Many projects have adopted OO models to facilitate integration [A$^+$91] and, recently, systems for the integration of sources with minimal structure have appeared [S$^+$93, PGMU95]. Futhermore, the DARPA Intelligent Integration of Information (I^3) research program is devoted to this problem.

However, as a consequence of the rapid development of prototype implementations in this area, the initial outcome of this program appears the production a new set of systems. While they can perform certain advanced information integration tasks, they cannot easily communicate with each other. Recently, a workshop was held on this topic at the University of Maryland in April, 1996 [BRU96], coming out with the proposal of a common data description and manipulation language which is a minimal subset of ODMG93 [Cat96]and covers relational systems.

Our approach to integration follows the TSIMMIS architecture[1]. This architecture is common to many data integration projects developed in the database area: *wrappers/translators* convert data into a common model while *mediators* [Wie92] combine, integrate or refine the data from the wrappers. The wrapper

*This research has been partially funded by the MURST 40 % Italian Project: 'Basi di Dati Evolute: Modelli, metodi e sistemi'.

Proceedings of the 4th KRDB Workshop
Athens, Greece, 30-August-1997

(F. Baader, M.A. Jeusfeld, W. Nutt, eds.)

http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-8/

[1] TSIMMIS is a joint project between Stanford (biblio references: http://db.stanford.edu) and the IBM Almaden Research Center whose goal is the development of tools that facilitate the rapid integration of heterogeneous textual sources that may include both structured and unstructured data [CMH$^+$94, GM$^+$95].

provides also a common query language for extracting informations. The translator converts also the queries over the common model into requests that the source can execute and the data extracted from the source into the common model. Languages to express queries and to specify mediators in a declarative style are provided. The possible bottlenecks of the above architecture are that an ad-hoc wrapper must be developed for any information source and implementing a mediator can be complicated and time-consuming. The main issues for these systems are:

- to provide a wrapper generator which can generate a wrapper based on a description of the conversion that need to take place for the received queries and the returned results;
- to automatically or semi-automatically generate mediators from high level descriptions of the information processing they need to do.

With respect to the above goals, our approach focuses on the second one, as we rely on a semantic high level language for the mediator description and on a general purpose computing procedure. We obtain the following benefits:

1. The language is an extension with rules of the structural part of the ODMG93 standard;
2. the language allows a declarative description of structures and translation rules;
3. the language is interpreted as a Description Logics and has an *open world* semantics approach;
4. tools for query optimization and consistency check of the mediator are available.

On the other hand, we require a "more cooperative" approach from the information sources, with respect to TSIMMIS: the schemas of the sources in the ODMG93 standard language must be provided. In summary, we propose that a source provide a description, in an adequate language, of its own information.

The outline of the paper is the following.

In Section 2, the `odlc` logics, a description logics developed for Object Oriented Databases, and the tool `ODBTOOLS` developed by the authors are briefly recalled.

Section 3 sketchs a description logics approach to the problems of data modelling and data querying in heterogeneous multiple sources of textual data. Furthermore, this 'semantic' approach is compared with the 'structural' approach of the TSIMMIS system.

Section 4 shows, by means of an example, our semantic approach to Mediators.

2 The `odlc` Description Logics and `ODBTOOLS`

Description Logics Languages (DLs) are concerned with only structural aspects; concepts roughly correspond to database classes (primitive concepts) and *views* (defined concepts) and are organized in inheritance taxonomies.

By exploiting defined concepts semantics of DLs, and, given a *type as set* semantics to concept descriptions, it is possible to provide reasoning techniques: to compute *subsumption* relations among concepts(i.e. "isa" relationships implied by concepts descriptions) and to detect *incoherent* (i.e. always empty) concepts. By means of DLs reasoning techiques, a view, can be automatically *classified* (i.e., its right place in an already existing taxonomy can be found) by determining the set of its most specific subsumer views (*subsumers*) and the set of its most generalized specialization views (*subsumees*).

`odlc` (Object Description Language with Constraints), derived from `odl`(Object Description Language) [BN94], is a description logics which represents the structural part of OODB data models (and of the standard data model *ODM* of ODMG93 [Cat96]) and adds (to ODMG93) the capability of expressing *integrity constraints rules.* Integrity constraints (IC) rules are *if then rules*, whose antecedent and consequent are OCDL virtual types (i.e. type descriptions expressing a set of sufficient and necessary conditions) allowing the declarative formulation of a relevant set of integrity constraints. For example, it is possible to express correlations between structural properties of the same class or sufficient conditions for populating subclasses of a given class.

Coherence checking and subsumption computation are also effective for query optimization. A query has the semantics of a virtual class, as it expresses a set of necessary and sufficient conditions. If we restrict the query language to the subset of queries expressible with the schema description language we can perform incoherence detection and subsumption computation for queries [BJNS94, BB97]. Note that, other research works state that in the general case the language for schema descriptions (i.e. classes) and queries (i.e. views) must be of different complexity [BDNS]. We agree with this position in general, but in the context of extraction and integration of textual heterogeneous data sources, we believe that *a highly expressive OO schema description language can be adopted as well as query language.* The choice of a simple query language (a significative restriction of the standard ODMG93 OQL) has been also recently made at the I^3 workshop on mediators language standards [BRU96].

Futhermore, queries referred to a single target class can be expressed as `odlc` types and DLs techniques can be exploited to produce the *semantic expansion* of a `odlc` query, which incorporates any possible restriction not present in the original query but *logically implied* by the query and the schema (classes, value types, IC rules). Following the approach of [SO89] for semantic query optimization and by exploiting subsumption computation to evaluate logical implication in [BBLS93, BBSV97], the semantic expansion of a query is performed.

A tool, say `ODBTOOLS` [BBSV97] (running demo at: http://sparc20.dsi.unimo.it) has been developed at the Dipartimento di Scienze dell'Ingegneria of the University of Modena to perform schema validation and query optimization in OODB. It is an open component for the input/ouput interface, compatible with the ODMG93, both for the schema definition language and the query language.

3 A Semantic Approach for Mediators: `odlc` Description Logics

The *TSIMMIS* approach towards *mediators* development is 'structural':

- OEM (the schema language), in fact, is a *self-describing model* where each data item has an associated descriptive label and *without a strong typing system*;
- *semantic informations are effectively encoded in the MSL (Mediator Specification Language) rules* that do the integration.

There are many projects following the 'structural approach' [DH86, $B^{+}86$]. Let us recall some fundamental arguments in favour of the 'structural approach' (considering *TSIMMIS* as a target system):

1. MSL and OEM can be seen as a form of first-order logic: rules are supported allowing the sharing of definitions of terms among components;
2. the generality and conciseness of OEM and MSL make the 'structural' approach a good candidate for the integration of widely heterogeneous and semi-structured information sources; this is an improvement, since:
    - in traditional data models, a client must be aware of the schema in order to pose a query, while here the structure of the information is discovered as queries are posed;
    - a conventional OO language breaks down in such a case, unless one defines an object class for every possible type of irregular object.

Let us argue that point 2 can be satisfied as well with a semantic 'approach' with a *weaker* class description notion.

- 2. revisited The adoption of an open world semantics for classes descriptions (i.e. tuple types in `odlc`) allows semi-structured data integration: objects of a class share a common minimal structure, but may have further additional properties.

Many other projects follow a 'semantic' approach [HM93]. This approach can be characterized as follows:

- for each source, meta-data, i.e. conceptual schema, must be available;
- semantic informations are encoded in the schema;
- a common data model as the basis for describing sharable informations must be available;
- partial or total schema unification is performed.

Let us introduce some fundamental arguments in favour of a 'semantic approach' based on conventional OO data models:

1. most research areas (programming languages, databases and artificial intelligence) take advantage of conventional OO models with strong type systems and including: classes, aggregation and inheritance hierarchies to model structural intensional knowledge and, often, methods to model behavioural knowledge;
2. a relevant effort has been devoted to develop OO standards: CORBA [VV.93] for object exchanging among heterogeneous systems; ODMG93 (including ODM model and ODL language for schema description; OQL language as query language) for object oriented databases [Cat96];
3. the schema nature of conventional OO models together with classification aggregation and generalization primitives allows to organize extensional knowledge;
4. the adoption of a *type as a set* semantics for a schema permits to check consistency of instances with respect to their descriptions;
5. semantic knowledge encoded into a schema permit to efficiently extract information.

By coupling a 'semantic approach', based on a description logics component (i.e. `ODBTOOLS`), a minimal ODMG93 standard interface and some features of *TSIMMIS*, it is possible to devise a powerful I^3 system (see the architecture in figure 1):

1. the standard ODM model and ODL language (ODMG93) as common data model and common data language are adopted both for sources and mediators;
2. the ODL language is extended to represent rules in analogy with *MSL*;
3. the ODL language is extended to represent QDTL;
4. a *minimal core language* which is a restriction of the object oriented query language OQL (from ODMG93) is adopted; in such a way, the language will accept also queries for relational databases[2];
5. a component based on description logics with interfaces for the above languages is adopted.

A mediator can be generated with the above I^3 system by introducing the following knowledge:

- describe the schemata of the sources to be integrated and the mediator schema in the ODL language;
- describe *query templates* in the *minimal core language*;
- describe the mediator rules.

Having `ODBTOOLS` available, the knowledge expressed in the standard languages above is automatically translated into `odlc` classes and the `odlc` incoherence detection and subsumption algorithms can be exploited in the following way:

- to perform data integration by exploiting mediator rules;
- to execute a query by determining the most efficient one among the supported subsuming queries.

A final remark: for sources supporting OODBMS or RDBMS, query templates, i.e. descriptions of the queries supported by a source, are not necessary, since the standard query languages are directly supported.

4 An example of the semantic approach to Mediators

In this section we sketch an example of the semantic approach to Mediators. As a first step, we will consider an example drawn from the TSIMMIS papers

[2]this choice is also suggested in the proposal for a standard in mediator languages [BRU96]

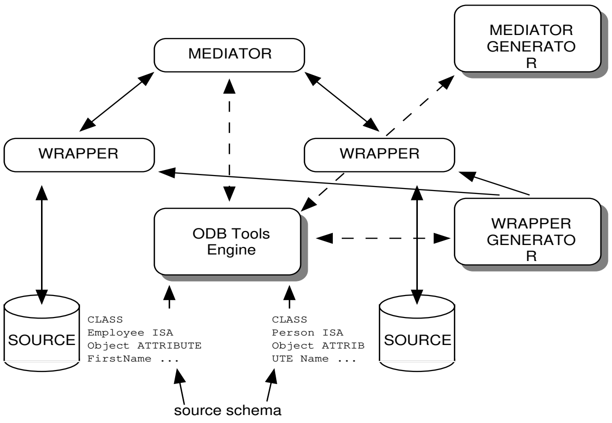

Figure 1: Architecture of an I^3 system

and show how it is translated into the ODL and ODM languages. Then we will show how the mediator rules can be expressed and we will give more hints on the expected services provided by `ODBTOOLS`. Figure 2 shows the OEM description of a piece of data, a professor and a student, deriving from a data source.

```
<&e1, employee, set, {&f1,&l1,&t1, &rep1}>
  <&f1, first_name, string, 'Joe'>
  <&l1, last_name, string, 'Chung'>
  <&t1, title, string, 'professor'>
  <&rep1, reports_to, string, 'John
Hennessy'>
'chairman'>
<&s3, student, set, {&f3,&l3,&y3}>
  <&f3, first_name, string, 'Pierre'>
  <&l3, last_name, string, 'Huyn'>
  <&y3, year, integer, 3>
```

Figure 2: **CS** objects in OEM

We propose that each source has its own schema, in ODL syntax, as it is shown in figure 3. In addition, a major feature of the OEM model is the extensibility of the schema, that is the ability to accept objects with new attributes. From this point of view, following the tradition of description logics, we can interpret the object schema as a sufficient condition and accept as a person any object showing *at least* the described properties. Analogously, figures 4 and 5 show a *whois* object in OEM syntax and its schema in ODL syntax.

Next step is the definition of the mediator providing the integrated view of the two sources. Figure 6 and 7 show the definition of the mediator MED in MSL and OQL respectively.

To obtain the integrated result, three major hypotheses have been made:

- the OQL definition relies on the additional fictitious class *decomp*, which is necessary to translate source attributes into integrated schema at-

```
interface cs_person:object  (
extent cs_persons):  persistent
{
  attribute String first_name;
  attribute String last_name;
}
interface employee:cs_person
( extent employees):  persistent
{
  attribute String title;
}
interface student:cs_person
( extent students):  persistent
{
  attribute integer year;
}
```

Figure 3: **CS** object schema in ODMG

```
<&p1, person, set, {&n1, &d1, &rel1,
&elem1}>
  <&n1, name, string, 'Joe Chung'>
  <&d1, dept, string, 'cs'>
  <&rel1, relation, string, 'employee'>
  <&elem1, e_mail, string, 'chung@cs'>
```

Figure 4: **whois** objects in OEM

tributes when no straightforward one to one mapping is possible: in this case, it maps a complete name string (i.e. blank separated) to/from a first name - last name pair;

- we assume that a method *class* is available to give the most specific class an object belongs to;
- we assume that a method *rest* is available to give the set of the attribute values other than those specified as arguments.

When the integrated source is to be queried, the query is mapped to queries on the sources, and the results are integrated by the mediator. If the sources do not have full dbms capabilities, but are able to answer only according to given *query templates* [PGMUG95], the subsumption procedure can be exploited in order to find out the query template which best covers the user query and thus obtain a query optimization. If, on the other hand, a source has dbms capabilities, the optimization can be executed taking into account the semantics of its schema.

To conclude, we showed a way to express wrappers and mediators in the ODMG model and, having available a set of tools for the schema verification and having available a set of tools for the vrification

```
 interface person:object
( extent persons):  persistent
{
  attribute String name;
  attribute String dept;
  attribute String relation;
  attribute String e-mail;
}
```

Figure 5: *whois* objects in ODMG

```
<&cp1, cs_person, set,
   {&mn1, &mrel1, &t1, &rep1, &elm1}>
  <&mn1, name, string, 'Joe Chung'>
  <&mrel1, relation, string, 'employee'>
  <&t1, title, string, 'professor'>
  <&rep1, reports_to, string, 'John
Hennesy'>
  <&elem1, e_mail, string, 'chung@cs'>
```

Figure 6: Object exported by MED

of ODMG schemas and the optimization of ODMG queries, we suggest to import these services in an architecture for the extraction of information from heterogeneous sources.

Among the many problems which still have to be solved, we can mention the computational characteristics of the algorithms dealing with both the schema definition and query languages, the comparison of effectiveness with respect to other possible choices and the language extensions necessary to reach a satisfactory expressiveness of the mediator and wrapper specification.

```
<cs_person {<name N> <rel R> Rest1 Rest2}>
        :- <person {<name N>
                <dept 'cs'> <relation R>
                | Rest1}>@whois
          AND decomp(N, LN, FN)
          AND <R {<first_name FN>
                <last_name LN>
                | Rest2}>@cs
External:
decomp(string,string,string)(bound,free,free)
  impl by name_to_lnfn
decomp(string,string,string)(free,bound,bound)
  impl by lnfn_to_name.
```

Figure 7: Mediator specification in MSL

```
select struct(name: y.name,
    relation: y.relation,
    rest1: x.rest(first_name,last_name),
    rest2: y.rest(name)
from x in cs_person,
     y in person,
     z in decomp
where y.name = z.name and
      x.first_name = z.first_name and
      x.last_name = z.last_name and
      y.name = z.name and
      x.class = y.relation
```

Figure 8: Mediator query in ODL-ODMG93

References

[A+91] R. Ahmed et al. The pegasus heterogeneous multidatabase system. *IEEE Computer*, 24:19–27, 1991.

[B+86] Y.J. Breibart et al. Database integration in a distributed heterogeneous database system. In *Proc. 2nd Intl IEEE Conf.on Data Engineering, Los Angeles, CA*, 1986.

[BB97] Domenico Beneventano and Sonia Bergamaschi. Incoherence and subsumption for recursive views and queries in object-oriented data models. *Data and Knowledge Engineering*, 21(3):217–252, February 1997.

[BBLS93] D. Beneventano, S. Bergamaschi, S. Lodi, and C. Sartori. Using subsumption in semantic query optimization. In A. Napoli, editor, *IJCAI Workshop on Object-Based Representation Systems*, pages 19–31, August 1993.

[BBSV97] Domenico Beneventano, Sonia Bergamaschi, Claudio Sartori, and Maurizio Vincini. ODB-qoptimizer: a tool for semantic query optimization in OODB. In *Int. Conference on Data Engineering - ICDE97*, 1997.

[BJNS94] M. Buchheit, M. A. Jeusfeld, W. Nutt, and M. Staudt. Subsumption between queries to object-oriented database. In *EDBT*, pages 348–353, 1994.

[BN94] S. Bergamaschi and B. Nebel. Acquisition and validation of complex object database schemata supporting multiple inheritance. *Applied Intelligence: The International Journal of Artificial Intelligence, Neural Networks and Complex Problem Solving Technologies*, 4:185–203, 1994.

[BDNS] M. Buchheit, F.M. Donini, W. Nutt, and A. Schaerf. Terminological Systems Revised: Terminology = Schema +Views. Working Notes of KRDB-94 Workshops. DFKI Report D-94-11.

[BRU96] P. Bunemann, L. Raschid, and J. Ullman. Mediator languages - a proposal for a standard. Technical report, University of Maryland, 1996. ftp://ftp.umiacs.umd.edu /pub/ONRrept/ medmodel96.ps.

[Cat96] R. G. G. Cattel. *The Object Database Standard - ODGM93*. Morgan Kaufmann, 1996.

[CMH+94] S. Chawathe, Garcia Molina, H., J. Hammer, K.Ireland, Y. Papakostantinou, J.Ullman, and J. Widom. The TSIMMIS project: Integration of heterogeneous information sources. In *IPSJ Conference, Tokyo, Japan*, 1994. ftp://db.stanford.edu /pub/chawathe/1994/ tsimmis-overview.ps.

[DH86] U. Dayal and H. Hwuang. View definition and generalization for databse integration in a multidatabase system. In *Proc. IEEE Workshop on Object-Oriented DBMS - Asilomar, CA*, 1986.

[GM+95] H. Garcia-Molina et al. The TSIMMIS approach to mediation: Data models and languages. In *NGITS workshop*, 1995. ftp://db.stanford.edu /pub/garcia/1995/tisimmis-models-languages.ps.

[HM93] J. Hammer and D. McLeod. An approach to resolving semantic heterogeneity in a federation of autonomous, heterogeneous database systems. *Intl Journal of Intelligent and Cooperative Information Systems*, 2:51–83, 1993.

[PGMU95] Y. Papakonstantinou, H. Garcia-Molina, and J. Ullman. Medmaker: A mediation system based on declarative specification. Technical report, Stanford University, 1995. ftp://db.stanford.edu /pub/papakonstantinou/1995/ medmaker.ps.

[PGMUG95] Y. Papakonstantinou, H. Garcia-Molina, J. Ullman, and Ashish Gupta. A query translation scheme for rapid implementation of wrappers. Technical report, Stanford University, 1995. ftp://db.stanford.edu /pub/papakonstantinou/1995/querytran-extended.ps.

[S+93] K. Shoens et al. The rufus system: Information organization for semistructured data. In *Proc. VLDB Conference - Dublin, Ireland*, 1993.

[SO89] S. Shenoy and M. Ozsoyoglu. Design and implementation of a semantic query optimizer. *IEEE Trans. Knowl. and Data Engineering*, 1(3):344–361, September 1989.

[VV.93] AA. VV. The common object request broker: Architecture and specification. Technical report, Object Request Broker Task Force, 1993. Revision 1.2, Draft 29, December.

[Wie92] G. Wiederhold. Mediators in the architecture of future information systems. *IEEE Computer*, 25:38–49, 1992.