SI-Web: A Web based interface for the MOMIS project[1]

D. Beneventano and S. Bergamaschi and D. Bianco, F. Guerra and M. Vincini

University of Modena and Reggio Emilia. Dipartimento di Ingegneria dell'Informazione,
Via Vignolese 905, 41100 Modena, Italy

d.bianco@biasoft.com
{beneventano.domenico, bergamaschi.sonia, guerra.francesco,
vincini.maurizio}@unimo.it

Abstract. The MOMIS project (Mediator envirOnment for Multiple Information Sources) developed in the past years allows the integration of data from structured and semi-structured data sources. SI-Designer (Source Integrator Designer) is a designer support tool implemented within the MOMIS project for semi-automatic integration of heterogeneous sources schemata. It is a java application where all modules involved are available as CORBA Object and interact using established IDL interfaces. The goal of this demonstration is to present a new tool: SI-Web (Source Integrator on Web), it offers the same features of SI-Designer but it has got the great advantage of being usable on Internet through a web browser.

1. Overview of the MOMIS system

Like other integration projects [1,8], MOMIS follows a "semantic approach" to information integration based on the conceptual schema, or metadata, of the information sources, and on the I3 architecture [6] (see Figure 1).
The system is composed by the following functional elements that communicates using the CORBA (OMG) standard:

- a common data model, ODMI3, which is defined according to the ODLI3 language, to describe source schemas for integration purposes. ODMI3 and ODLI3 have been defined in MOMIS as subset of the corresponding ones in ODMG, following the proposal for a standard mediator language developed by the I3/POB working group [5]. In addition, ODL I3 introduces new constructors to support the semantic integration process;
- Wrappers, placed over each sources, translate metadata descriptions of the sources into the common ODL I3 representation, translate (reformulate) a global query expressed in the OQLI3 query language into queries expressed in the sources languages and export query result data set;

[1] This work was partially supported by MIUR co-funded project D2I.

- Mediator, which is composed of two modules: the SI-Designer and the Query Manager (QM). The SI-Designer module processes and integrates ODLI3 descriptions received from wrappers to derive the integrated representation of the information sources. The QM module performs query processing and optimization. The QM generates OQLI3 queries to be sent to wrappers starting from each query posed by the user on the Global Schema. QM automatically generates the translation of the query into a corresponding set of sub-queries for the sources and synthesizes a unified global answer for the user.

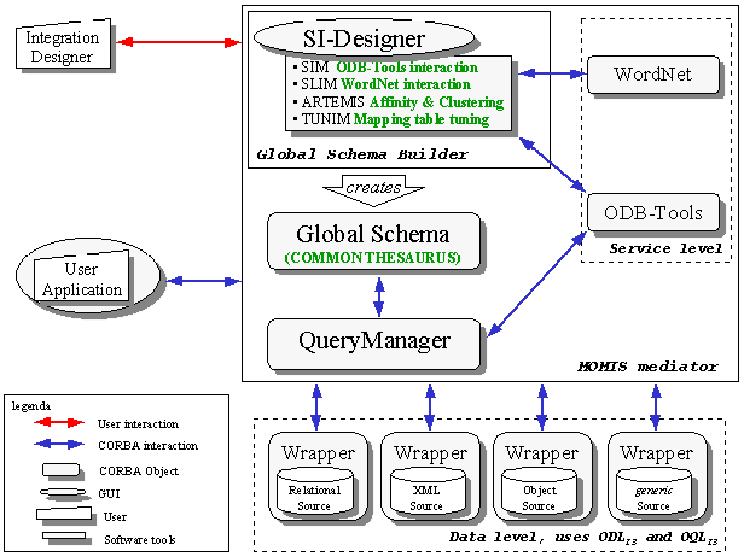

Fig. 1. The MOMIS Architecture

2. SI-Designer Tool

As described above, the integration process consists of various steps actually implemented in separate module. SI-Designer is a framework that represents a unified solution for the overall integration process. SI-Designer provides the designer with a graphical interface to reach the Global Virtual View, relating to each integration step a specific interaction with a software module.
In particular, the SI-Designer performs these steps:

- Source acquisition: in this phase, the user can select the sources to be integrated. A wrapper performs the translation from the source description model into ODL description model. This step involves SAM module.
- Intensional relationships definition: in this phase, new relationships, schema derived, by interacting with SIM module and ODB-Tool system [9], lexicon derived, by interacting with the WordNet [7] lexical database, and designer

supplied are added to the Common Thesaurus (in Figure 3 an example of relationships is shown).

- Extensional relationships definition: Extensional relationships are defined by the interaction with the integration designer. These relationships are exploited to detect extensionally overlapping classes.
- Clustering: in this phase, based on the knowledge carried in the Common Thesaurus, by exploiting ARTEMIS module, global classes are created.
- Mapping table tuning: for each global class generated in the previous phase, the user can modify the Global Virtual View proposed automatically from the system. The final step of the integration process provides the export of the Global Virtual View into a XML DTD, by adding the appropriate XML TAGs to represent the mapping table relationships. The use of XML in the definition of the Global Virtual View lets to use MOMIS infrastructure with other open integration information system by the interchange of XML data files. In addition, the Common Thesaurus is translated into XML file, so that MOMIS may provide a shared ontology that can be used by different semantic ontology languages.

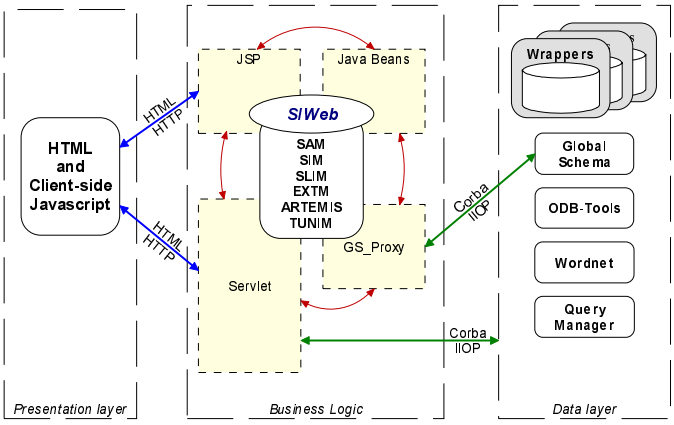

Fig. 2. The SI-Web Architecture

3. SI-Web tool

SI-Web has been developed in order to reproduce the same user interface of SI-Designer on a website. This solution has been adopted to mitigate the impact of a new interface and to reduce the costs of a new training of SI-Designer users.
A three tiers architecture has been used to achieve this goal, in fact this type of architecture has the following advantages:

- High availability and scalability of the system through many servers that can share the workload;
- Good response time with any kind of client as the major part of the application runs on the server side;

- Client platform independence as languages different from HTML may be supported to allow connections from any kind of client (i.e. WML for mobile phone etc.);
- Easy upgrade of the application as it resides on a server (it is not necessary to update each client).

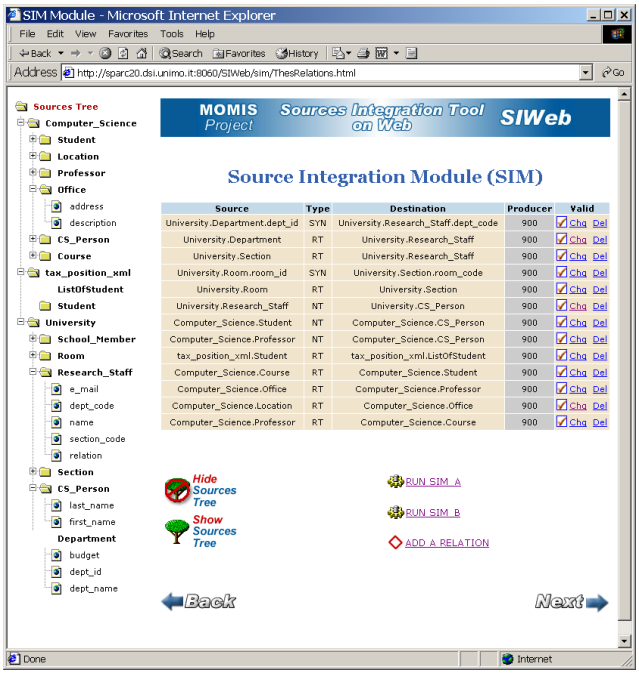

Fig. 3. SI-Web intensional and extensional relationships

4. SI-Web architecture

As shown in the figure 2, we are going to describe how the three tiers have been implemented:

- Presentation layer: it consists in HTML code and Javascript dynamically generated and it does elementary client side functions such as input validation. Since the integration system has a sequential structure, we decided to substitute all form panels with links to the previous and next steps;
- Business logic: the real application resides in this tier and it has been implemented using Servlet, Java Server Pages and Javabeans. This tier has two functions: on one hand it generates the GUI (Graphical User Interface) producing dynamically the HTML sent to the client, on the other hand it handles the input received from the

client and it communicates by CORBA with the objects which are part of MOMIS architecture;

- Data layer: it is made of CORBA objects, which are the "MOMIS kernel". These objects offer a wide range of services such as ODB-Tools, WordNet, Wrappers and some others integration services.

Figure 3 shows one of the integration process steps. SI-Web produces the same results in a layout that aims to look like SI-Designer GUI as much as possible because both applications call (through CORBA) the same MOMIS objects. Therefore, the "business logic" of SI-Web produces dynamically the HTML code sent to the client browser.

References

[1] Y. Arens, C.Y. Chee, C. Hsu, and C. A. Knoblock, 1993. 'Retrieving and integrating data from multiple information sources'. International Journal of Intelligent and
Cooperative Information Systems, 2(2):127-158, 1993.

[2] D. Beneventano, S. Bergamaschi, S. Castano, A. Corni, R. Guidetti, G. Malvezzi, M. Melchiori, and M. Vincini, 2000. 'Information integration: The MOMIS project demonstration'. Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt, pages 611-614. Morgan Kaufmann, 2000.

[3] S. Bergamaschi, S. Castano, and M. Vincini, 1999. 'Semantic integration of semi-structured and structured data sources'. SIGMOD Records, 28(1), March 1999.

[4] S. Bergamaschi, S. Castano, D. Beneventano, and M. Vincini, 2001. 'Semantic integration of heterogenous information sources'. Journal of Data and Knowledge Engineering, 36(3):215-249, 2001.

[5] P. Buneman, L. Raschid, and J. Ullman, 1996. 'Mediator languages - a proposal for a standard', April 1996. Available at ftp://ftp.umiacs.umd.edu/pub/ONRrept/- medmodel96.ps.

[6] R. Hull and R. King et al., 1995. 'Arpa I3 reference architecture', 1995. Available at http://www.isse.gmu.edu/I3Arch/index.html.

[7] A. G. Miller, 1995. 'Wordnet: A lexical database for English', Communications of the ACM, 38(11):39-41, 1995.

[8] M. T. Roth and P. Scharz, 1997. 'Don't scrap it, wrap it! a wrapper architecture for legacy data sources'. In Proc. of the 23rd Int. Conf. on Very Large Databases, Athens, Greece, 1997.

[9] D. Beneventano, S. Bergamaschi, C. Sartori, and M. Vincini. ODB-QOPTIMIZER: A tool for semantic query optimization in oodb. In Int. Conference on Data Engineering - ICDE97, 1997.