Project FP6-2005-IST-5-034980

STASIS

SofTware for Ambient Semantic Interoperable Services

Deliverable DX6

MOMIS-STASIS: Specification and Implementation

Workpackage WP6 – Validation
Task T6.2 – Final STASIS Validation

Version	3.01	**Date**	2009-09-07	**Classification**	Public	**Status**	STASIS Approved

Abstract

STASIS (SofTware for Ambient Semantic Interoperable Services) is a Research and Development project sponsored under the European Commission's 6th Framework programme as well as its projects members – www.stasis-project.net. Its objective is for Research, Development and Validation of open, web Services based, distributed semantic services for SME empowerment within the Automotive, Furniture and other sectors. It commenced September 1st 2006 and lasts for 3 years until August 2009 with a total budget of €4M. 12 Partners are involved including Commercial Companies (TIE, iSoft) Academics (Universities of Sunderland, Oldenburg, Modena & Reggio Emilia, Tsinghua) and User Organisations (AIDIMA, Mariner, Shanghai Sunline, Foton, TANET, ZF Friedrichshafen AG) and these are led by the managing partner TIE. Partners are spread across Europe and China.

This deliverable is an informal deliverable that won't be officially reviewed but is intended to introduce the STASIS audience to the MOMIS-STASIS framework for Ontology-Based Data Integration. More in details, the purpose of this deliverable is to describe the specification and the implementations activities for the realization of a prototype that implements the MOMIS-STASIS approach for Ontology-Based Data Integration.

Authors STASIS Consortium (www.stasis-project.net)
Responsible Editor: UOM, Domenico Beneventano
Additional Editors: UOM, Serena Sorrentino and Laura Po and Sara Quattrini

Executive Summary

This document describes the specification and implementations activities of the MOMIS-STASIS approach for Ontology-Based Data Integration.

MOMIS (Mediator envirOnment for Multiple Information Sources) is a framework developed by UOM to perform information extraction and integration from both structured and semi-structured data sources. With the MOMIS-STASIS approach for Ontology-Based Data Integration an important synergy was established between STASIS and the MOMIS framework.

From a scientific point of view, the MOMIS-STASIS approach has given positive results; some papers were submitted and accepted for presentation at international level:

- The first International Workshop on Interoperability through Semantic Data and Service Integration, which was held on June 25th 2009, during SEBD '09 (17th Italian Symposium on Advanced Database Systems), Camogli (Genova), Italy
- The 2009 International Workshop on Semantic Computing and Multimedia Systems IEEE-SCMS 2009, Held in conjunction with the Third IEEE International Conference on Semantic Computing (ICSC 2009), Berkeley, CA, USA - September 14-16, 2009

Moreover, this MOMIS-STASIS activity is compliant and coherent with the exploitation intention of UOM, stated in the DOW: "Academic University of Modena is very active in research areas such as semantic web, mediator systems ... UOM expects that a joint research initiative involving all these different themes represents a breakthrough in the area and will obtain relevant scientific results. On the basis of these results it will be possible to study and develop new technologies for integrated search and negotiation systems, thus further empowering the UOM competence in this field."

This deliverable is an informal deliverable that will not be officially reviewed but is intended to introduce the STASIS audience to the MOMIS-STASIS framework for Ontology-Based Data Integration. The purpose of this deliverable is to describe the specification and the implementations activities for the realization of a prototype that implements the MOMIS-STASIS approach for Ontology-Based Data Integration.

STASIS PARTNERS

Table of Contents

Table of Contents

1 Background

The purpose of this section is to introduce the:

- STASIS Project
- Purpose, scope and context of this deliverable
- Intended audience for the deliverable
- Document Structure
- External References

1.1 STASIS Project

STASIS (SofTware for Ambient Semantic Interoperable Services) is a Research and Development project sponsored under the European Commission's 6th Framework programme as well as its projects members – www.stasis-project.net. Its objective is for Research, Development and Validation of open, web services based, distributed semantic services for SME empowerment within the Automotive, Furniture and other sectors. It commenced September 1st 2006 and lasts for 3 years until August 2009 with a total budget of €4M. 12 Partners are involved including Commercial Companies (TIE, iSOFT) Academics (Universities of Sunderland, Oldenburg, Modena & Reggio Emilia, Tsinghua) and User Organisations (AIDIMA, Mariner, Shanghai Sunline, Foton, TANET, ZF Friedrichshafen AG) and these are led by the managing partner TIE. Partners are spread across Europe and China.

1.2 Deliverable purpose, scope and context

This deliverable is an informal deliverable that will not be officially reviewed but is intended to introduce the STASIS audience to the MOMIS-STASIS framework for Ontology-Based Data Integration.

In STASIS, mappings of schemata are mainly used to perform translation, i.e. to transform (map) data and documents from one format to another format. Indeed, this is the primary use of mappings. On the other hand, as discussed in literature, mapping among schemata are also useful for another important interoperability task, i.e. data integration [Abiteboul1999,Kalfoglou2005, Zamboulis2008].

In particular, in [Hai2007], the author states that semantic correspondences between elements of metadata structures are of key importance for interoperability and data integration in numerous applications, such as data warehousing, integration of web-sources, message mapping in E-business, and ontology alignment on the Semantic Web.

To this end, this deliverable proposes the so-called MOMIS-STASIS approach for data integration, where mappings among schemas produced by the STASIS framework are used in the MOMIS data integration system. This approach was published and discussed to the following conferences:

- The first International Workshop on Interoperability through Semantic Data and Service Integration, which was held on June 25th 2009, during SEBD ’09 (17th Italian Symposium on Advanced Database Systems), Camogli (Genova), Italy
- The 2009 International Workshop on Semantic Computing and Multimedia Systems IEEE-SCMS 2009, Held in conjunction with the Third IEEE International Conference on Semantic Computing (ICSC 2009), Berkeley, CA, USA - September 14-16, 2009

The purpose of this deliverable is to describe the specification and implementations activities of the MOMIS-STASIS approach for data integration, an approach that given positive results from a scientific point of view. Moreover, the deliverable introduces the demo of the MOMIS-STASIS approach in the context of the main STASIS Demo scenario.

What is the main benefit of the MOMIS-STASIS activity for the STASIS framework? One of the key aspects of the STASIS framework is a federated P2P repository where all elements including source schemata and mappings, are stored. With the MOMIS-STASIS approach there is the great opportunity to exploit this quantity of source schemata and mappings available in the STASIS Network for an important task in the field of the semantic interoperability - ie the integration of these source schemata.

From a technical point of view, the scope of this deliverable is also to demonstrate that the output produced by the STASIS framework can be easily imported in another semantic tool, such as MOMIS, based on a different data model.

1.3 Audience

The intended audience includes:

- User project partners to establish a basis for the validation
- Project partners and External parties interested in the possible use of STASIS application

1.4 Document Structure

This document is structured as follows:

- **Section 2: Introduction**
  Describes the MOMIS-STASIS approach for data integration. In the first part of this a very simple example of mapping among schemata is proposed, then, these mappings are used to perform data translation and data integration, by comparing these two different ways to use mappings produced by the STASIS framework. In the second part of the section the MOMIS-STASIS approach and a demo scenario for this approach are introduced.

- **Section 3: Specifications**
  The functional specification and the technical specification of the MOMIS-STASIS framework are introduced in section 3
- **Section 4: Implementation**
  A detailed description of the implementation with particular attention to the GUI interface is shown in section 4
- **Section 5: Conclusions**
  The conclusions highlights the benefits of the MOMIS-STASIS approach by comparing it with the MOMIS framework
- **Appendix A**
  Contains the MOMIS-STASIS approach for Ontology-Based Data Integration as published and discussed in the first International Workshop on Interoperability through Semantic Data and Service Integration, and in the 2009 International Workshop on Semantic Computing and Multimedia Systems.

1.5 References

This document is dependent on the following primary references:

- STASIS Deliverable D7.2 “Demonstration Activities”
- STASIS Description of Work (DOW) [STASIS DOW]
- STASIS Glossary [STASIS DX1 Glossary]

2 Introduction

In STASIS mappings among schemata are mainly used to perform translation, i.e. to transform/map data and documents from a format to another format. This is surely the primary use of mappings and will be also used to demonstrate the STASIS framework as discussed in Deliverable D7.2.

On the other hand, as discussed in literature, mappings among schemata are also useful for another important interoperability task, i.e., data integration [Abiteboul1999,Kalfoglou2005,Zamboulis2008].In particular, in [Hai2007], the author states that semantic correspondences between elements of metadata structures are of key importance for interoperability and data integration in numerous applications, such as data warehousing, integration of web-sources, message mapping in E-business, and ontology alignment on the Semantic Web.

In this section, a very simple example of mapping among schemata (subsection 2.1) is introduced; for the sake of simplicity and without loss of generality, two simple relational tables are considered. Then, in subsection 2.2, these mappings are used to perform data translation and data integration, by comparing these two different ways to use mappings produced by the STASIS framework. Then, in subsection 2.3 the MOMIS-STASIS approach is introduced and, finally, in subsection 2.4 a demo scenario for this approach is shown.

2.1 A simple mapping example

The Figure 1 shows mappings produced by the STASIS framework between two relational tables.

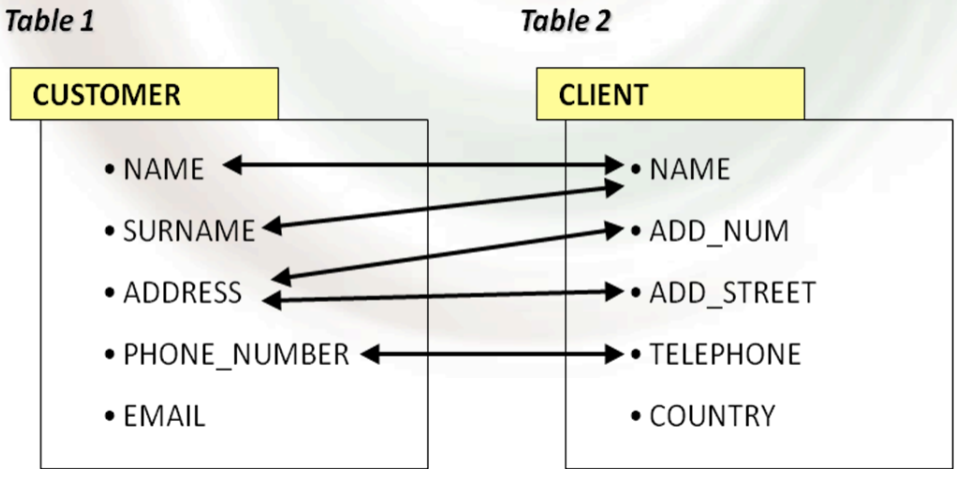

Figure 1 - Mapping produced by the STASIS framework

Some instances of these two tables are in Figure 1.

CUSTOMER

NAME	SURNAME	ADDRESS	PHONE_NUMBER	EMAIL
Tom	Cooper	341, Old Paradise Street	+44 20 7623 2335	tom.cooper@gmail.com
Jennifer	Baker	56, GladStone Street	+44 20 7823 1880	j.baker@thm.uk
...	...	...	...	...

CLIENT

NAME	ADD_NUM	ADD_STREET	TELEPHONE	CITY
Tom Cooper	341	Old Paradise Street	NULL	London
Joe Jolly	789	Keyworth Street	+44 20 7629 5713	London
...	...	...	...	

Figure 2 - Instance level of the tables CUSTOMER and CLIENT

2.2 Translation vs Integration

In the translation process, data from *a source table* is extracted, transformed then loaded into *a target table*; **integration** involves combining data of different sources and providing users with a unified view of these data. In our example, the integration process applied to the two tables CLIENT and CUSTOMER aims to generate a new *integrated global table* and mappings among this new *global* table and the original *local* tables CLIENT and CUSTOMER.

In the MOMIS Integration system the structure of the global table, i.e. its *global attributes*, and mappings among these global attributes and the *local attributes* are represented in the so-called *Mapping Table* (see Figure 2) where:

- The first column represents the global attributes
- The other columns contain the local attributes of the local tables belonging to the global table

The Mapping Table is a table where the columns represent the local tables belonging to the global table G and whose rows represent the global attributes of G. An element MT[GA][L] represents the set of local attributes of the local table L which are mapped onto the global attribute GA.

In this way it is possible to express that the global attribute NAME is mapped into NAME and SURNAME for the local table CUSTOMER and into NAME for the local table CLIENT. The global attribute ADDRESS is mapped into the local attribute ADDRESS for the local table CUSTOMER and into ADD_STREET and ADD_NUM for the local table CLIENT. The global attribute PHONE_NUMBER is mapped into the local attribute PHONE_NUMBER for the local table

CUSTOMER and into the local attribute TELEPHONE for the local table CLIENT. Finally, COUNTRY and EMAIL global attribute are mapped respectively only to the CLIENT and CUSTOMER local table.

GLOBAL TABEL - CUSTOMER

GLOBAL ATTRIBUTE	LOCAL ATTRIBUTE CUSTOMER	LOCAL ATTRIBUTE CLIENT
NAME	NAME, SURNAME	NAME
ADDRESS	ADDRESS	ADD_STREET, ADD_NUM
CITY	NULL	CITY
PHONE_NUMBER	PHONE_NUMBER	TELEPHONE
EMAIL	EMAIL	NULL

Figure 3 - Mapping Table of the Global Table

A **global integrated schema** is a set of global tables; for each global table the related mapping table is defined.

The global schema allows a user to pose a query and receive a unique answer, transparently from the involved sources; ie the MOMIS Query Manager is the coordinated set of functions which take an incoming query on a global schema, define a decomposition of the query according to the mapping tables, send the sub queries to the data sources, collect their answers, perform any residual filtering as necessary, and finally deliver the answer to the requesting user (see www.dbgroup.unimo.it/Momis for more details and publications about MOMIS).

2.3 MOMIS-STASIS approach

The integration process of two or more local sources with the related local schemas, in a global schema is a two step process:

1. **Clusterization of semantic related tables**: Individuation of tables of different schemas which represent *semantic related information* and thus which can be integrated in the same global class. In the above example this task is trivial; in a real scenario, with dozens of schemata to be integrated and where each schema may contain hundreds of tables, this is a complex task.
2. **Construction of Mapping Tables**: For each cluster of tables identified at in step 1, a Mapping Table is generated. Also this task may be a complex task since each table may contain dozens or hundreds of attributes.

Ultimately, in a real scenario, the construction of the global schema with the related Mapping Tables may be a complex task.

The key aspect of the MOMIS approach for data integration is as follows: **given mappings among elements of the local schemas**, the MOMIS system provides methods and tools to automate either the two above steps of the integration process. In particular, in the Clusterization step, the affinity of two tables is established by means of affinity coefficients based on table structures and on the given mappings between table attributes. Then, tables with affinity are grouped together in clusters using hierarchical clustering techniques. The goal is to identify the tables that have to be integrated since they describe the same or semantically related information.

The integration designer may interactively refine and complete the proposed integration results, i.e., the mappings which have been automatically created by the system can be fine tuned. In particular, for a global attribute mapped on more than one local attributes the designer needs to define *mapping functions* (in our example, NAME is mapped into NAME and SURNAME, then the designer can use the *conjunction* mapping function: NAME + SURNAME).

The key aspect of the MOMIS-STASIS approach is as follows: *the mappings among elements of the local schemas, needed for the MOMIS integration process, are produced by the STASIS framework.*

One of the key aspects of the STASIS framework is a federated pure P2P repository, where all elements including source schemata and mappings are stored and are searched with the STASIS DESKTOP APPLICATION in order to reuse them; during the mapping process, all information is continually stored in the STASIS Network. With the MOMIS-STASIS approach there is a great opportunity to exploit this quantity of source schemata and mappings available in the STASIS Network for an important task in the field of the semantic interoperability, that is the integration of these source schemata.

2.4 Translation vs Integration: Demo scenario

In this section, first is introduced the “classical” STASIS demo scenario as discussed in Deliverable D7.2. Then, to highlight the differences between translation and integration, this translation demo scenario is compared with a scenario where integration is performed.

2.4.1 Translation: Demo scenario

In STASIS, people start by importing their own format into the STASIS Neutral Format (SNF). When importing, SSEs are created automatically. As an alternative, people can create/connect SSEs manually. In the next step, people are using the STASIS comparator for mapping their SSEs to the SSEs of a business partner. Overall, the conceptual result – and therefore the data/knowledge that is available – looks like in Figure 4.
The SIF format is taking this knowledge and combines it into one file with three ‘sub-files’ inside:

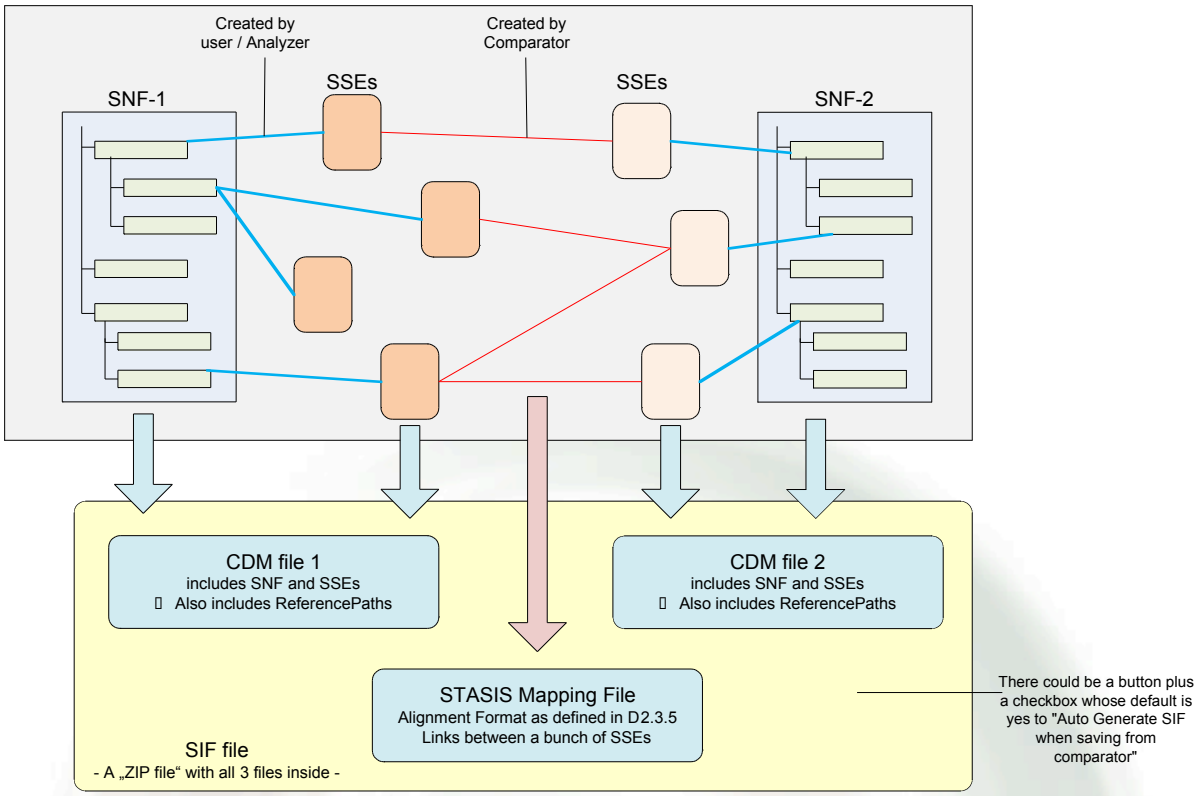

Figure 4 - Conceptual result of the STASIS framework

In the STASIS Demo scenario proposed in Deliverable D7.2, based on this SIF file, some implementations are created (see Figure 5(1) and Figure 5(2)):

- A “Mini-Processor”; this is a small and independent application that takes a SIF file and create a minimalistic mapping between Excel files
- A converter will be written that takes the SIF file and converts it to an XSLT.
  → Demonstrator implementation only: it is not needed to support all XSLT syntax. Only simply 1:1 mappings (e.g. no Choices, etc.)

With the MOMIS-STASIS framework, an importer will be developed and implemented, that takes the SIF file and converts it to an input for the MOMIS framework (see Figure 5(3)).

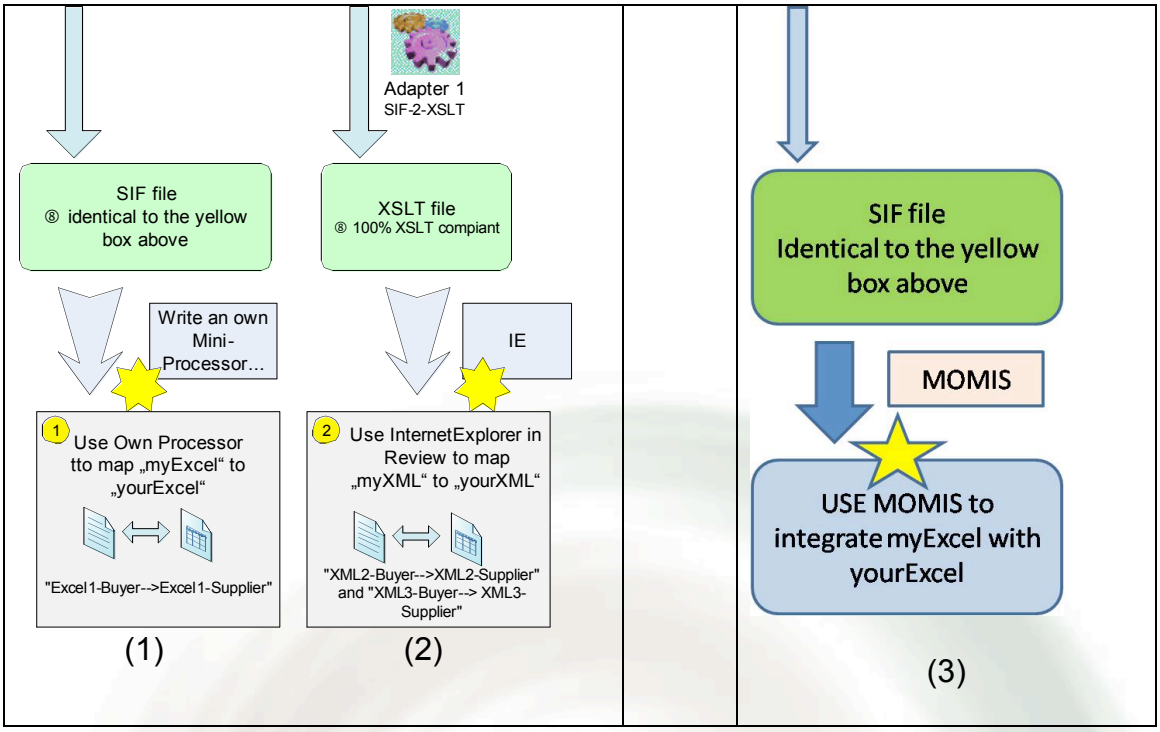

Figure 5 - Implementations based on the SIF file

At a more general and schematic level, what it is possible to show with the MOMIS-STASIS framework is presented in Figure 6, where show is the STASIS Demo scenario and what the MOMIS-STASIS framework adds to this demo scenario.

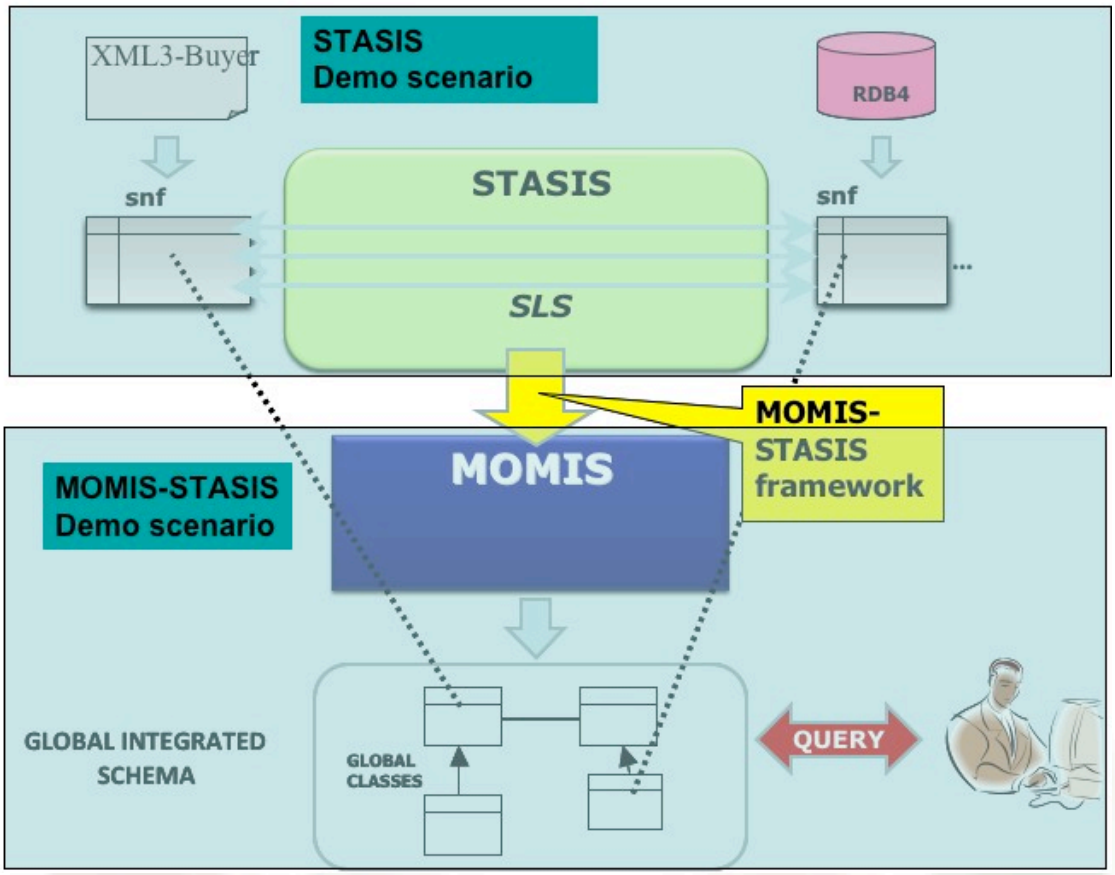

Figure 6 - MOMIS-STASIS Demo scenario

2.5 The MOMIS-STASIS scenario

In this section, the demo of the MOMIS-STASIS approach in the context of the main STASIS demo scenario proposed in Deliverable D7.2 is introduced. The starting point of the MOMIS-STASIS demo are the results, the output produced in one ore more scenarios of the main STASIS Demo. One of the main advantages of the SIF files is that having this export formats allows us the real-world applicability of STASIS. More precisely it is possible to show how to use SIF as input to a complex real-world mapper. In our MOMIS-STASIS demo this complex real-world mapper is a real data-integration tool: MOMIS.

In the MOMIS-STASIS scenario, the SIF file produced by STASIS by considering two RDBs sources (Source and Destination) is given as an input at the MOMIS system, the an global and integrated schema is automatically produced and this global schema can be queried.

Input:

The input of this scenario is the SIF file coming from two RDB; RDB4 and a “relation version of” XML3-Supplier.xsd or XML3-Buyer.xsd[1] are considered.

Output:

The output is a global and integrated schema of the two input schemata. This output will be shown with the GUI of the MOMIS system. Moreover, with another panel of the MOMIS system, the user can query (in SQL or with a GUI interface) the global schema.

[1] An automatic translation of an XML source into a RDB database was realized in MOMIS, but this is in a very prototype state, never tested. For this reason, it has been preferred to manually obtain a new RDB source

3 MOMIS-STASIS: Specifications

This section contains the functional specification and the technical specification of the MOMIS-STASIS framework.

3.1 Functional specifications

This section defines the functional specification of the MOMIS-STASIS framework. It is directly based on:

- CDM model, the STASIS Path Language (SPL) and the SIF output produced by STASIS;
- ODLI3 model and the relationships (SYN, BT, RT) used in MOMIS and it represents the main specification for the MOMIS-STASIS implementations.

As shown in **[STASIS: Demo Scenarios]**, the output of STASIS is essentially a ZIP file with the file extension “SIF”. It will contain three files:

1. **source.snf**
   contains the SNF information of the imported source schema as well as its SSEs and their connection to the SNF
2. **destination.snf**
   same as source.snf but for the destination
3. **mapping.af**
   the mapping between the source SSEs and destination SSEs in the Alignment Format

As stated in [STASIS: Demo Scenarios], the STASIS Path Language (SPL) is necessary so that a translator has the necessary structural information to be able to access/enter data in an instance (content file) of an original schema. With the SPL it is possible to point to syntactical locations in messages in the SIF file so translators can be fed. The SPL is defined with reference to the CDM of STASIS.

The first question to solve is the following: in the MOMIS-STASIS approach are all these three files necessary or is the mapping file (mapping.af) only sufficient?
To answer this question, a comparison among the CDM model used in STASIS and the ODLI^3 model used in MOMIS is needed and a precise definition of correspondences between the concepts of these two models. In this way it is possible to define if and how a SPL can individuate, in a unique way, an element in an ODLI3 schema. This action is fundamental in order to translate the link produced by STASIS and contained in the SIF file in the MOMIS framework. This comparison is discussed in section 3.1.1, and the conclusion of this comparison is that the Mapping file mapping.af **(**the mapping between the source SSEs and destination SSEs in the Alignment Format) is sufficient to translate SLS in the STASIS format into mapping of the MOMIS framework.

This result will be used in the Technical Specification of the MOMIS-STASIS framework

3.1.1 Comparison between the CDM model (STASIS) and the ODLI3 model (MOMIS)

The comparison between the CDM model and the ODLI3 model used in MOMIS is illustrated with reference to the examples of the demo scenario. The first step is to obtain the ODLI3 version of a schema used in the demo scenario. To this end XML3-Buyer.xsd is loaded in MOMIS and the following schema in ODLII3 is (automatically) generated.

```
Class ${OrderType}  {
      attribute string ${PurchaseOrderNumber};
      attribute string ${SupplierName};
      attribute OrderType_mgroup_1 ${OrderType_mgroup_1};
      attribute OrderType_mgroup_2 ${OrderType_mgroup_2};
}

Class ${OrderType_mgroup_1} {
      attribute set <Address_type> ${Address};
}

Class ${Address_type} {
      attribute null ${AddressType};
      attribute string ${StreetAddress};
      attribute string ${CountryCode};
   }

Class ${OrderType_mgroup_2} {
      attribute set <LineItemType> ${LineItem};
   }

Class ${LineItemType} {
 attribute long ${ItemNumber};
 attribute string ${PartName};
 attribute Price_type ${Price};
 attribute LineItemType_mgroup_1 ${LineItemType_mgroup_1};
   }

Class ${Price_type} {
   attribute any ${Amount};
   attribute any ${CurrencyCode};
}

Class ${LineItemType_mgroup_1} {
       attribute set <Schedule_type> ${Schedule};
}

Class ${Schedule_type} {
  attribute float ${Quantity};
  attribute date ${ExpectedDeliveryDate}; }
```

For a comparison with the CDM of STASIS, the XML3-Supplier.snf has been considered; an abstract representation of XML3-Supplier.xsd, obtained by the STASIS application, is shown in
Figure 7.

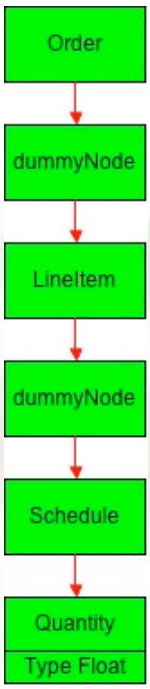

Figure 7 - An abstract representation of XML3-Supplier.xsd

In this figure, it is possible to see the correspondences with the ODLI3 schema, in particular the use of DUMMY nodes in correspondence of the ODLI3 element set: to say that an ORDER has many ADDRESS, in ODLI3 it is necessary to insert a dummy node "OrderType_mgroup_1"; something to similar is performed in STASIS where these DUMMY nodes, are inserted when it is necessary.

Let us consider the following reference path contained in the mapping.af file.

```
<cdm:SchemaReferencePath>/Order/LineItem/Schedule/Quan
tity</cdm:SchemaReferencePath>
```

…and let us consider the portion of the ODLI3 schema related to this path

```
Class ${OrderType}  {
...
      attribute OrderType_mgroup_2 ${OrderType_mgroup_2};
}


Class ${OrderType_mgroup_2} {
      attribute set <LineItemType> ${LineItem};
   }
```

```
Class ${LineItemType} {
 ...
 attribute LineItemType_mgroup_1 ${LineItemType_mgroup_1};
   }

Class ${LineItemType_mgroup_1} {
       attribute set <Schedule_type> ${Schedule};
}

Class ${Schedule_type} {
  attribute float ${Quantity};
  attribute date ${ExpectedDeliveryDate}; }
```

It can be demonstrated that, also in the ODLI3 schema, the above reference path identifies in an unique way the `Quantity` attribute of a `Schedule`. Starting from the above considerations, the Mapping file **mapping.af (**the mapping between the source SSEs and destination SSEs in the Alignment Format) (contained in the SIF file) is sufficient to translate SLS in the STASIS format into mapping of the MOMIS framework.

3.1.2 Comparison between SLS (STASIS) and MOMIS relationships

In the MOMIS-STASIS approach the following links (SLS) between semantic entities of two schemas are considered

- Equivalence (EQUIV) (= in the .af file), with the related weight
- More general (SUP) (> in the .af file) , with the related weight
- Less general (SUB) (< in the .af file) with the related weight

…and the following correspondences between these SLS' and links (SYN,BT,RT) used in MOMIS are used:

- EQUIV corresponds to SYN (synonymous)
- SUP corresponds to BT (broader)
- SUB corresponds to NT (narrower)

Moreover, in order to consider the weight associated with a SLS in STASIS: an EQUIV SLS will be translated to a SYN link into MOMIS only if its weight is greater than a given threshold X, otherwise it will be translated to a to a *weak link* in MOMIS called RT (Related terms). The default value for X will be established on the basis of some experiments; the user can change this default value. The same holds for the other SLS. The Translation from SLS of STASIS to links in MOMIS is shown in Table 1.

STASIS		**MOMIS**
Type	**Weight**	**Type**
EQUIV (=)	>= X	SYN (synonymous)
EQUIV (=)	< X	RT (related)
SUB (<)	>= Y	NT (narrower)
SUB (<)	< Y	RT (related)

SUP (>)	>= Y	BT (broader)
SUP (>)	< Y	RT (related)

Table 1 – Translation from SLS of STASIS to links in MOMIS

3.2 Technical specifications

This section is the Technical Specification of the MOMIS-STASIS framework. It specifies the technical implementation needed to realize the functionality which has been identified above, in particular to realize the Translation from SLS of STASIS to MOMIS relationships as specified in the previous section.

Once selected the file mapping.af, a set of Java classes will take care to translate the STASIS mappings in a format understandable to MOMIS.
This process will be realized in several steps:

1. The construction of a parser written in Java, that will be able, for each STASIS mapping, to recognized and extract from the Mapping.af file the following elements: the two schema elements between which the mapping is relevant (and their reference path) and the local source to which they belong; the type of the mapping (i.e. SYN or BT etc.) and the weight associated to the mapping.
2. After that, it is necessary to recover the correspondent ODLI3 elements for each mapping. To do this Java classes are implemented. Given a path and a ODLI3 schema these classes will be able to recognized which ODLI3 elements correspond to the reference path previously extracted. However, this phase requires further study in order to understand whether it is necessary to implement different methods on the bases of the local source type (e.g. relational database or xml schema etc.).

   In particular, for an XML schema source file, it should be studied in what way it is needed to navigate the ODLI3 on the basis of the reference paths. In the case of relational sources, the identification of the correspondent ODLI3 element on the basis of the reference path is quite simple, since the path will composed by at more two levels: for the level 1 an ODLI3 interface (class) will be referred to, while in the case of level 2, an ODLI3 attribute will be referred to.
3. Once this information is obtained, a java class needs to be developed that will be able to create the MOMIS semantic relationship (corresponding to the STASIS mapping) for the new STASIS-MOMIS Common Thesaurus. In such a case java methods and classes already present in the MOMIS system can be reused.

4 MOMIS-STASIS: GUI implementation

To develop the tool, the idea was to provide, by using the GUI of MOMIS, a modality through which the user can use the mappings provided by STASIS, instead of using the traditional MOMIS relationships discovery process. This implies the modification of the MOMIS GUI. However, the new MOMIS-STASIS tool will be a new tool independent from the MOMIS system.

To understand how the MOMIS-STASIS approach will be implemented, start by considering the actual/standard MOMIS GUI as shown in Figure 8, where the main steps of the MOMIS integration process are also indicated:

Schema Loading: Similar to STASIS

MOMIS link Generation: This part implements the MOMIS links generations among schemata, mainly based on the annotation re WordNet. There are three tabs:

- *ALA (Automatic Lexical Annotation)*: automatic annotation w.r.t. the WordNet Lexical ontology
- *Annotation* : manual annotation by the user (mainly is a specialization of the automatic annotation produced by ALA)
- *Common Thesaurus* : MOMIS link generation (based on annotation) and management

Global Schema Generation: MOMIS techniques to cluster element and then to automatically generate a global schema

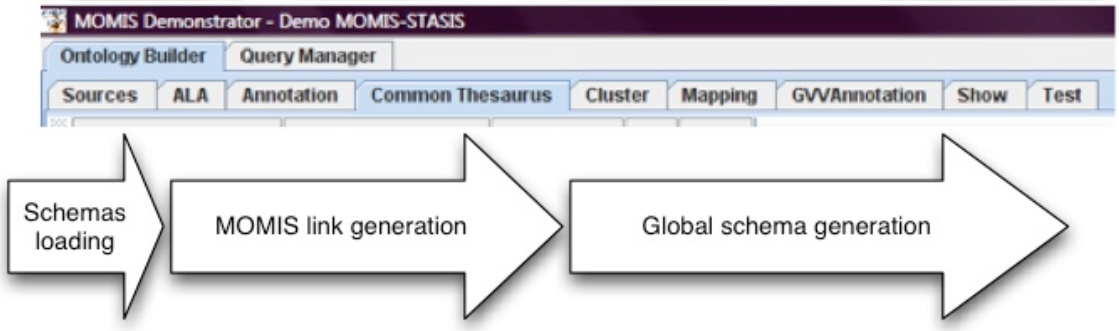

Figure 8 - Actual MOMIS GUI

In the MOMIS-STASIS approach the second step of the integration process, i.e. the MOMIS link Generation, is skipped since references to the input schemas and links between these input schemata are obtained by STASIS by means of the SIF file. For this reason, by using the MOMIS-STASIS system, all the TABS related to link Generation will be not present in the GUI – specifically tabs

"ALA", "Annotation" and "Common Thesaurus" which are used in MOMIS to perform the semantic relationships discovery process.

The first proposal to implement the MOMIS-STASIS approach is based on the modification of the MOMIS GUI by adding a new tab. To give an intuitive idea about this, consider the above Figure 8 with a screenshot of the current MOMIS GUI. The idea is to insert a new tab (called for example STASIS-MOMIS) through which the user can interact with the MOMIS-STASIS system.

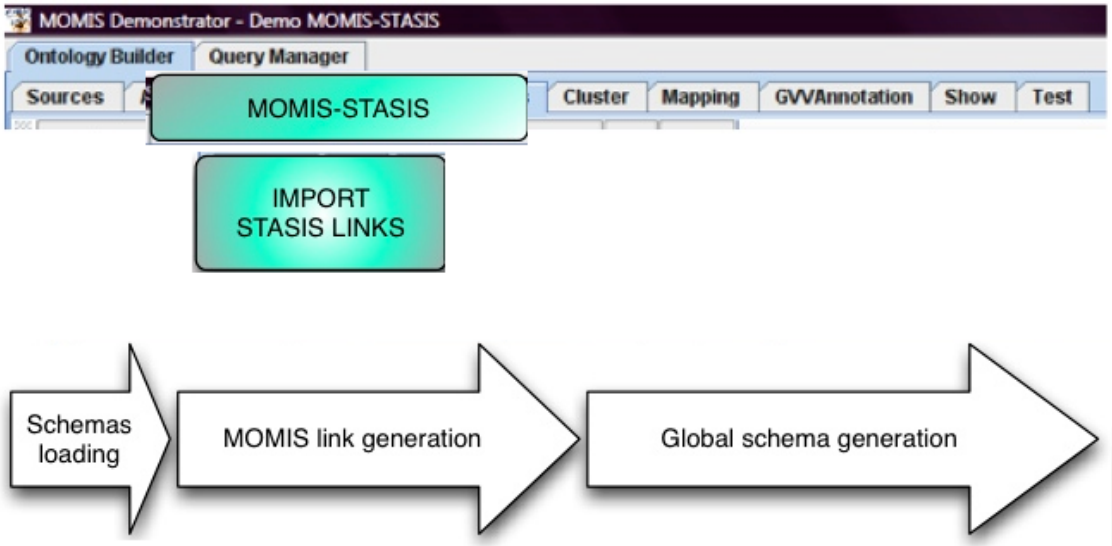

Figure 9 - The MOMIS-STASIS GUI

A new TAB called MOMIS-STASIS is introduced. It contains the following button:

IMPORT STASIS links:

To import in MOMIS the STASIS Mapping.af file, to the new tab a functionally is added that allows to the user to select from the file directory the file Mapping.af that contains the links to use during the data integration process.

During this phase the MOMIS-STASIS system automatically converts the STASIS semantic links in MOMIS relationships, as discussed in section 3.1.2 (The conversion is realized in a transparent way with respect to the user). The conversion is performed on the basis of the Table 1 that shows the conversion rules, to translate every type STASIS mapping in the correspondent MOMIS relationship (SYN, BT etc.). For example, a Synonym STASIS relationship ("=") with an associated weight over a given X value, will be translate in a SYN MOMIS relationship. The table contains default weights but the user may modify these weights by using the GUI.

Visualization of the translated links

Finally, the MOMIS relationships obtained starting from the STASIS mapping, will be shown in a table in panel of the MOMIS-STASIS tab: a table similar to the Table 2 will show the final translated relationships collected in the new MOMI-STASIS Common Thesaurus.

SSE1(source)	SSE2 (Destination)	Type	PRODUCER
A	B	SYN	STASIS
A1	B1	RT	STASIS
A2	B2	BT	STASIS
A3	B3	RT	STASIS
A4	B4	BT	STASIS
A5	B5	RT	STASIS

Table 2 - Visualization of the translated links

This new tab will be very similar to the current Common Thesaurus tab (shown in above), but it will be modified in order to manage the MOMIS-STASIS mappings. All the other MOMIS tabs (Cluster, Mapping etc.) will be unchanged.

5 Conclusion

MOMIS (Mediator envirOnment for Multiple Information Sources) is a framework developed by UOM to perform information extraction and integration from both structured and semi-structured data sources.

With the MOMIS-STASIS approach for Ontology-Based Data Integration an important synergy was established between STASIS and the MOMIS framework, a related research framework in the field of semantic interoperability.

From a scientific point of view, the MOMIS-STASIS approach has given positive results: some papers were submitted and accepted for presentation at international level.

Moreover, this MOMIS-STASIS activity is compliant and coherent with the exploitation intention of UOM, stated in the DOW: “Academic University of Modena is very active in research areas such as semantic web, mediator systems ... UOM expects that a joint research initiative involving all these different themes represents a breakthrough in the area and will obtain relevant scientific results. On the basis of these results it will be possible to study and develop new technologies for integrated search and negotiation systems, thus further empowering the UOM competence in this field.“

A prototype that implements the MOMIS-STASIS approach was realized and a demo of the MOMIS-STASIS was introduced in the context of the main STASIS Demo scenario of deliverable 7.2.

Appendix A - The MOMIS-STASIS approach for Data Integration

To obtain a self-contained document, this appendix contains the MOMIS-STASIS approach for Ontology-Based Data Integration as published and discussed to the following conferences:

- The first International Workshop on Interoperability through Semantic Data and Service Integration, which was held on June 25th 2009, during SEBD '09 (17th Italian Symposium on Advanced Database Systems), Camogli (Genova), Italy
- The 2009 International Workshop on Semantic Computing and Multimedia Systems IEEE-SCMS 2009, Held in conjunction with the Third IEEE International Conference on Semantic Computing (ICSC 2009), Berkeley, CA, USA - September 14-16, 2009

A.1 INTRODUCTION

Data integration is the problem of combining data residing at distributed heterogeneous sources, and providing the user with a unified view of these data; a common and important scenario in data integration are structured or semi-structure data sources described by a schema. The core of data integration is solving the correspondences (find the right matches) among elements from different data sources schemata. The problem of designing Data Integration Systems is important in current real world applications, and is characterized by a number of issues that are interesting from a theoretical point of view [14]. Integration System are usually characterized by a classical wrapper/mediator architecture [21] based on a Global Virtual Schema (Global Virtual View -GVV) and a set of data sources. The data sources contain the real data, while the GVV provides a reconciled, integrated, and virtual view of the underlying sources. Modelling the mappings among sources and the GVV is a crucial aspect. Two basic approaches for specifying the mapping in a Data Integration System have been proposed in the literature: Local-As-View (LAV), and Global-As-View (GAV), respectively [13], [19].

MOMIS (Mediator EnvirOnment for Multiple Information Sources) is a Data Integration System which performs information extraction and integration from both structured and semi-structured data sources by following the GAV approach [8], [7]. Information integration is performed in a semi-automatic way, by exploiting the knowledge in a Common Thesaurus (defined by the framework) and descriptions of source schemas with a combination of clustering techniques and Description Logics. This integration process gives rise to a virtual integrated view of the underlying sources for which mapping rules and integrity constraints are specified to handle heterogeneity.

An ontology is an explicit specification of a conceptualization [12]. An ontology defines a set of representational primitives with which to model a domain of knowledge or discourse, and provides a shared vocabulary, which can be used to model a domain that is, the type of objects and/or concepts that exist, and their properties and relations. Ontologies offer a direction towards solving the interoperability problems brought about by semantic obstacles, such as the obstacles related to the definitions of terms and classes.

Ontologies can be used in an integration task to describe the semantics of the information sources and to make the contents explicit [20]. With respect to the integration of data sources, they can be used for the identification and association of semantically corresponding information concepts.

In [20], three different approaches of how to employ the ontologies for the explicit description of the information source semantics are identified: *single ontology approaches*, *multiple ontologies approaches* and *hybrid approaches*. *Single ontology approaches* use one global ontology providing a shared vocabulary for the specification of the semantics: all data sources are related to one global ontology. In *multiple ontology approaches*, each information source is described by its own ontology and mappings between the ontologies are defined: these inter-ontology mappings identify semantically corresponding terms of different source ontologies, e.g. which terms are semantically equal or similar. In *hybrid approaches* similar to multiple ontology approaches the semantics of each source is described by its own ontology, but in order to make the source ontologies comparable to each other they are built upon one global shared vocabulary which contains basic terms of a domain [20].

With respect to the above classification, the MOMIS Data Integration System uses a single ontology approach, where the lexical ontology WordNet [16] is used as a shared vocabulary for the specification of the semantics of data sources and for the identification and association of semantically corresponding information concepts. The main reason of this choice is that, by using a lexical ontology as WordNet (which it is characterized by a wide network of semantic relationships between concepts), the annotation of data sources elements can be performed in a semi-automatic way by using Word Sense Disambiguation techniques.

The STASIS IST project (www.stasis-project.net) is a Research and Development project sponsored under the EC 6th Framework programme. It aims to enable SMEs and enterprises to fully participate in the Economy, by offering semantic services and applications based on the open SEEM registry and repository network. The goal of the STASIS project is to create a comprehensive application suite which allows enterprises to simplify the mapping process between data schemas, by providing an easy to use GUI, allowing users to identify semantic elements in an easy way [2], [1].

Moreover, in the STASIS project, a general framework to perform Ontology-driven Semantic Mapping has been proposed, where the identification of

mappings between concepts of different schemas is based on the schemas annotation with respect to ontologies [5].

In [6] this framework has been further elaborated and it has been applied to the context of products and services catalogues. In the STASIS project OWL is used as language to include in the framework generic external ontologies.

This paper describes an approach to combine the MOMIS and STASIS frameworks in order to obtain an effective Global Schema Generation approach for Ontology-Based Data Integration. The proposal is based on the extension of the MOMIS system by using the Ontology-driven Semantic Mapping framework developed in STASIS in order to address the following points:

- Enabling the MOMIS system to employ *generic* OWL ontologies, with respect to the limitation of using only the WordNet lexical ontology;
- Enabling the MOMIS system to exploit a *multiple ontology* approach with respect to the actual *single ontology* approach;
- Developing a new method to compute semantic mapping among source schemas in the MOMIS system.

A.2 ONTOLOGY-BASED DATA INTEGRATION: THE MOMIS-STASIS APPROACH

This section describes our approach to use the Ontology-driven Semantic Mapping framework performed by STASIS for a different goal, i.e., during in the Global Schema Generation process performed by the MOMIS system. Intuitively, with the Ontology-driven Semantic Mapping framework we may perform in the Data Integration System the annotation of data sources elements with respect to generic ontologies (expressed in OWL), by eliminating in this way the MOMIS limitation to use only the lexical ontology WordNet. Moreover, we introduce in the MOMIS system a *multiple ontology* approach with respect to the actual *single ontology* approach. In the following, we will refer to this new approach as the MOMIS-STASIS approach.

The MOMIS-STASIS approach is shown in Figure 10. It can be divided into two macro-steps: STASIS: Semantic Link Generation (shown in Figure 10-a) and MOMIS : Global Schema Generation (shown in Figure 10-b).

A.2.1 STASIS: Semantic Link Generation

As stated in [2], [1] the key aspect of the STASIS framework, which distinguishes it from most existing semantic mapping approaches, is to provide an easy to use GUI, allowing users to identify semantic elements in an easy way. Once this identification has been performed STASIS lets users map their semantic entities to those of their business partners where possible assisted by STASIS. This allows users to create mappings in a more natural way by considering the meaning of elements rather than their syntactical structure. Moreover, all mappings that have been created by STASIS, as well as all semantic entities, are managed in a distributed registry and repository network.

This gives STASIS another significant advantage over traditional mapping creation tools as STASIS may reuse all mappings. This allows STASIS to make some intelligent mapping suggestions by reusing mapping information from earlier semantic links.

Besides the semantic links explicitly provided by the user, an Ontology-driven Semantic Mapping approach, for the STASIS framework, has been proposed [5]. The mappings between semantic entities being used in different schemas can be achieved based on annotations linking the semantic entities with some concepts being part of an ontology. In [6], this framework has been further elaborated and it has been applied to the context of products and services catalogues. An overview of the process for Ontology-driven Semantic Mapping Discovery is given in Figure 10-a. It can be summed up into 3 steps (each step number is correspondingly represented in figure): (1) obtaining a neutral schema representation, (2) local source annotation, and (3) semantic mapping discovery.

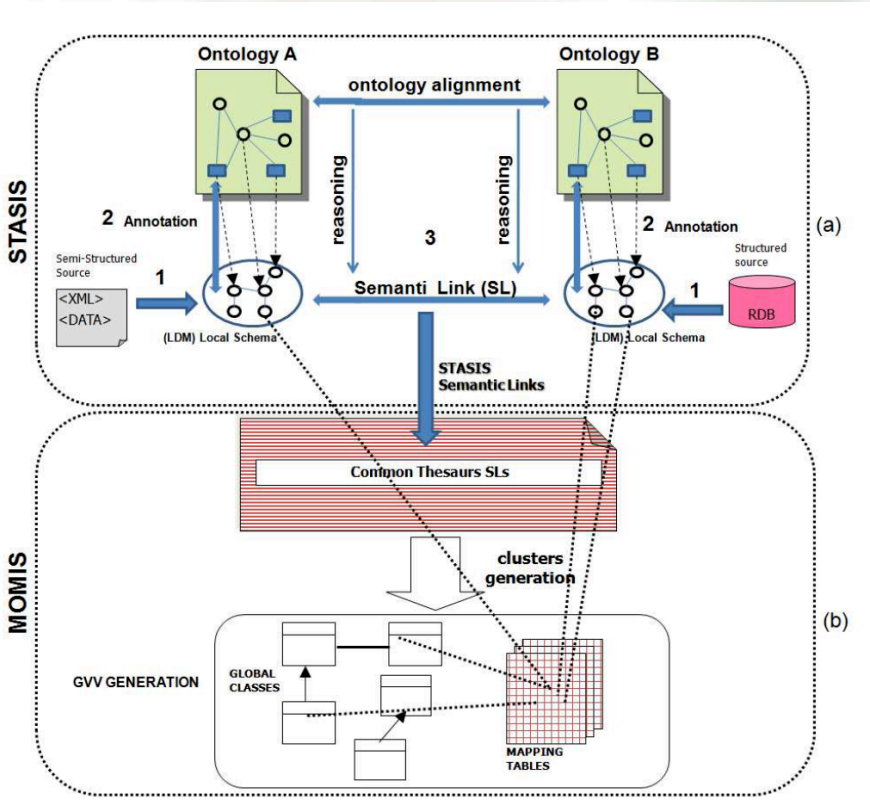

Figure 10 - The MOMIS-STASIS approach for Ontology-Based Data Integration

Step 1. Obtaining a neutral schema representation

As sketched in Figure 10-a, the STASIS framework works on a neutral representation, which abstracts from the specific syntax and data model of a particular schema definition; therefore, all the structural and semi-structural local sources first need to be expressed in a neutral format. The neutral representation is obtained by describing the local schemas through a unified data model called Logical Data Model (LDM). For the purpose of this paper, we

abstract from the specific features of LDM and we consider that this model contains common aspects of most semantic data models: it allows the representation of *classes* (or concepts) i.e. unary predicates over individuals, *relationships* (or object properties) i.e. binary predicates relating individuals, and *attributes* (or data-type properties) i.e. binary predicates relating individuals with values such as integers and strings; classes are organized in the familiar *is-a* hierarchy. *Classes*, *relationships* and *attributes* are called *semantic entities*.

Step 2. Local source annotation
The proposed mapping process identifies mappings between semantic entities through a "reasoning" with respect to aligned ontologies. Semantics of the data is captured by some kind of *semantic correspondences* between the database schema and ontologies. For this purpose the semantic entities need to be annotated with respect to one or more ontologies.

More formally, an *annotation element* is a 4-tuple $< ID, SE, R, concept >$ where ID is a unique identifier of the given annotation element; SE is a semantic entity of the schema; $concept$ is a concept of the ontology; R specifies the semantic relationship which may hold between SE and $concept$. The following semantic relationships between semantic entities and the concepts of the ontology are used: equivalence (AR_EQUIV); more general (AR_SUP); less general (AR_SUB); disjointness (AR_DISJ).

Actually within the STASIS framework are implemented only simple automatic annotation techniques, e.g. the "name-based technique" where the annotation between a semantic entity and a ontology concept is discovered by comparing only the strings of their names. The main drawback of this automatic technique is due to the existence of *synonyms* (when different words are used to name the same entities, e.g. "Last Name" and "Surname") and *homonyms* (when the same words is used to name different entities, e.g. "peer" has a sense "equal" as well as another sense "member of nobility") [10]. For these reason the designer have to manually refine the annotations in order to capture the semantics associated to each entities. In Section A.3 Future Work a preliminary idea to overcome this limitation is described.

Step 3. Semantic mapping discovery
Based on the annotation made with respect to the ontologies and on the logic relationships identified between these aligned ontologies, reasoning can identify correspondences among the semantic entities and support the mapping process. Given two schemas S1 and S2, and assuming that Ontology A and Ontology B are the reference ontologies which have been used to annotate the content of S1 and S2 respectively, given a mapping between Ontology A and Ontology B which provides a correspondence between concepts and relationships in the two ontologies, a semantic mapping between the annotated schemas S1 and S2 is derived. The following semantic mappings between entities of two source schemas (called *semantic link*-SL) can be discovered: equivalence (EQUIV); more general (SUP); less general (SUB); disjointness (DISJ); this definition is based on the general framework proposed in [11].

More formally, an SL is a 4-tuple *< ID, semantic_entity1, R, semantic_entity2 >,* where *ID* is a unique identifier of the given mapping element; *semantic_entity1* is an entity of the first local schema; *R* specifies the semantic relationship which may hold between *semantic_entity1* and *semantic_entity2*; *semantic_entity2* is an entity of the second local schema.
An application example of the Ontology Driven Semantic Mapping approach is described in Section A.2.3 Example other examples can be found in [6].

A.2.2 MOMIS: Global Schema Generation

In the MOMIS Data Integration System, information integration is performed by exploiting the semantic links among source schemas and using clustering techniques. Given a set of data sources it is thus possible to synthesize -in a semiautomatic way -a Global Schema (called *Global Virtual View* -GVV) and the mappings among the local source schemas and the GVV [8], [7].

In the MOMIS System, semantic links among source schemas are mostly derived with lexicon techniques based on the lexical annotation with respect to WordNet; then, all these semantic links are collected in a Common Thesaurus. In this paper we consider as semantic links among source schemas the semantic links defined with the STASIS framework; in other words, we consider as input of the GVV generation process the *Common Thesaurus SLs* generated by the STASIS framework. An overview of this GVV generation process is given in Figure 10-b.

In the GVV generation process, *local* classes describing semantically related concepts in different source sources are clusterized in the same *global* class of the GVV and mappings among this global class and its local classes and defined. More precisely, exploiting the Common Thesaurus SLs and the local sources schemas, our approach generates a GVV consisting of a set of global classes, plus a Mapping Table (MT) for each global class, which contains the mappings to connect the global attributes of each global class with the local sources' attributes.

A MT is a table where the columns represent the local classes belonging to the global class G and whose rows represent the global attributes of G. An element MT [GA][L] represents the set of local attributes of the local source L which are mapped onto the global attribute GA; an example of this process will be shown in next section. The integration designer may interactively refine and complete the proposed integration results; in particular, the mappings that have been automatically created by the system can be fine tuned.

The GVV is the intensional representation of the information provided by the Integration System; the next step is to specify how such an intensional representation relates to the local sources managed by the Integration System. MOMIS follows a Global-As-View (GAV) approach, then the GVV is designed to be a view over the local sources: each class of the GVV is characterized in

terms of a view over its local classes. On the basis of this view, a query posed by a user with respect to the global class can be rewritten as an equivalent set of queries (local queries) expressed on the local classes. The local query answers are then merged exploiting reconciliation techniques and proposed to the user.

The definition of the view associated to a global class and the related querying problem are out of the scope of this paper; for a complete description of the methodology to build and query the GVV see [8], [7].

A.2.3 Example

As a simple example let us consider two relational local sources L1 and L2 , where each schema contains a relation describing purchase orders:

```
L1: PURCHASE_ORDER(ORDERID, BILLING_ADDRESS, DELIVERY_ADDRESS, DATE)
L2: ORDER(NUMBER, CUSTOMER_LOCATION, YEAR, MONTH, DAY)
```

In the following, we will described step by step the application of the MOMIS-STASIS approach on these two sources.

STASIS: Semantic Link Generation

Step 1. Obtaining a neutral schema representation

During this step the local sources L1 and L2 are translated in the neutral representation and are represented in LDM data model; for a complete and formal description of a such representation see [5], where a similar example was discussed. As said before, for the purpose of this paper, we consider that the local schema L1 contains a class PURCHASE ORDER with attributes ORDERID, BILLING ADDRESS, DELIVERY ADDRESS, DATE.

In this way L1.PURCHASE ORDER, L1.PURCHASE ORDER.BILLING ADDRESS, L1.PURCHASE ORDER.DELIVERY ADDRESS etc. are semantic entities. In the same way the local schema L2 contains a class ORDER with attributes NUMBER, CUSTOMER LOCATION, YEAR, MONTH, DAY.

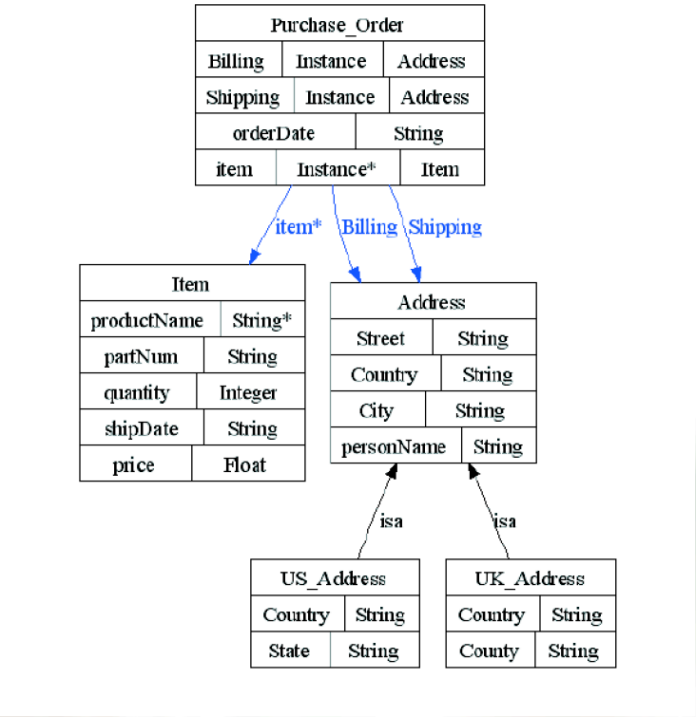

Figure 11 - The ontology of Purchase order

Step 2. Local Source Annotation
For the sake of simplicity we consider the annotation of schemas and the derivation of mappings with respect to a single common ontology ("Ontology-based schema mapping with a single common ontology" scenario considered in [5]).

Let us give some examples of annotations of the above schemas with respect to the Purchase Order Ontology shown in Figure 11. In the examples the identifier ID is omitted and a concept C of the ontology is denoted by "O:C". In a *simple annotation* the concept O:C is a primitive concept or a primitive role of the ontology (e.g. the class O:ADDRESS or the property O:BILLING). In a *complex annotation* the concept O:C is obtained by using the OWL language constructs (e.g. "O:ADDRESS and BILLING-1.Purchase Order" where BILLING-1 denotes the inverse of the property O:BILLING).

The following are examples of simple annotations:

```
(L1.PURCHASE_ORDER.BILLING_ADDRESS, AR_EQUIV, O:ADDRESS)
```
and
```
(L1.PURCHASE_ORDER.BILLING_ADDRESS, AR_EQUIV, O:BILLING).
```
These annotations are automatically discovered by applying the automatic "name-based" technique (see Section 2.1). However, as this technique does not consider the semantics associated to each entities, the following annotation
```
(L2.ORDER.CUSTOME_LOCATION, AR_EQUIV, O:ADDRESS)
```

is not discovered: the entities CUSTOME LOCATION and the concept ADDRESS have complete different names but, in this context, they have the same senses. In Section 3 a preliminary idea to overcome this problem is described.

An example of complex annotation is:

```
(L1.PURCHASE_ORDER.DELIVERY_ADDRESS, AR_EQUIV, O:Address and Shipping-
1.Purchase_Order)
```

which can be considered as a refinement by the designer of the above simple annotations to state that the address in the PURCHASE ORDER table is the "address of the Shipping in a Purchase Order".

Other examples of complex annotations are:

```
(L1.PURCHASE_ORDER.BILLING_ADDRESS, AR_EQUIV, O:Address and Billing-
1.Purchase_Order)
```

where is explicitly declared by the designer to state that the address in the PURCHASE ORDER table is the "address of the Billing in a Purchase Order".

```
(L2.ORDER.CUSTOMER_LOCATION, AR_EQUIV, O:Address and Shipping-
1.Purchase_Order)
```

where is explicitly declared by the designer to state that the address in the ORDER table is the "address of the Shipping in a Purchase Order".

Moreover, the designer supplies also the annotations with respect to the ontology for the semantic entities L1.PURCHASE ORDER.ORDERID, L1.PURCHASE ORDER.DATE and L2.ORDER.NUMBER, L2.ORDER.YEAR, L2.ORDER.MONTH, L2.ORDER.DAY.

Step 3. Semantic mapping discovery
From the previous annotations, for example, the following semantic link is derived:

```
(L2.ORDER.CUSTOMER_LOCATION, EQUIV,L1.PURCHASE_ORDER.DELIVERY_ADDRESS)
```

while no semantic link among CUSTOMER LOCATION and BILLING ADDRESS is generated.

MOMIS: Global Schema Generation
Given the set of semantic links described above and collected in the Common Thesaurus, the GVV is automatically generated and the classes describing the same or semantically related concepts in different sources are identified and clusterized in the same global class. Moreover, the Mapping Table shown in Table I is automatically created by the MOMISSTASIS approach. The global class ORDER is mapped to the local class ORDER of the L1 source and to the local class PURCHASE ORDER of the L2 source. The NUMBER, DATE and CUSTOMER ADDRESS global attributes are mapped to both the sources, the BILLING ADDRESS global attribute is mapped only to the L2 source.

Global attributes *ORDER*	Local attributes *ORDER*	Local attributes *PURCHASE ORDER*

NUMBER	NUMBER	ORDER ID
DATE	YEAR,MONTH,DAY	DATE
CUSTOMER	CUSTOMER	DELIVERY
LOCATION	LOCATION	ADDRESS
BILLING ADDRESS	*NULL*	BILLING ADDRESS

Table 3 – Mapping table example

A.3 Future Work

One of the main advantage of the proposed approach is an accurate annotation of the schemas that produces more reliable relationships among semantic entities. The relationships among semantic entities are then exploited in order to obtain a more effective integration process. On the other hand, this more accurate annotation has the disadvantage that is currently performed manually by the integration designer.
In this work, we describe only a preliminary idea to overcome the problem of manual annotation, which will be the main subject of our future research.

Several works about automatic annotation are proposed in literature but only a few of them are applied in the context of schemas/ontologies matching discovery. In [15], [9], where we introduced a mapping technique based on Lexical Knowledge Extraction: first, an Automatic Lexical Annotation method is applied to annotate, with respect to WordNet, schemas/ontologies elements then lexical relationships are extracted based on such annotations.

Moreover, in [17] a way to apply our Automatic Lexical Annotation method to the SCARLET matcher [4], is presented. SCARLET is a technique for discovering relations between two concepts by making use of online available ontologies. The matcher can discover semantic relations by reusing knowledge declared within a single ontology or by combining knowledge contained in several ontologies.

By applying Automatic Lexical Annotation based on WordNet, the matcher validates the discovered mapping by exploring the semantics of the terms involved in the matching process.

Starting from these works, we agree that the WordNet lexical ontology can be used to improve the annotation phase of the Ontology Driven Semantic Mapping process. The strength of a lexical ontology like WordNet is the presence of a wide network of semantic relationships among the different words meanings, which represent a key element for automatic annotation techniques.

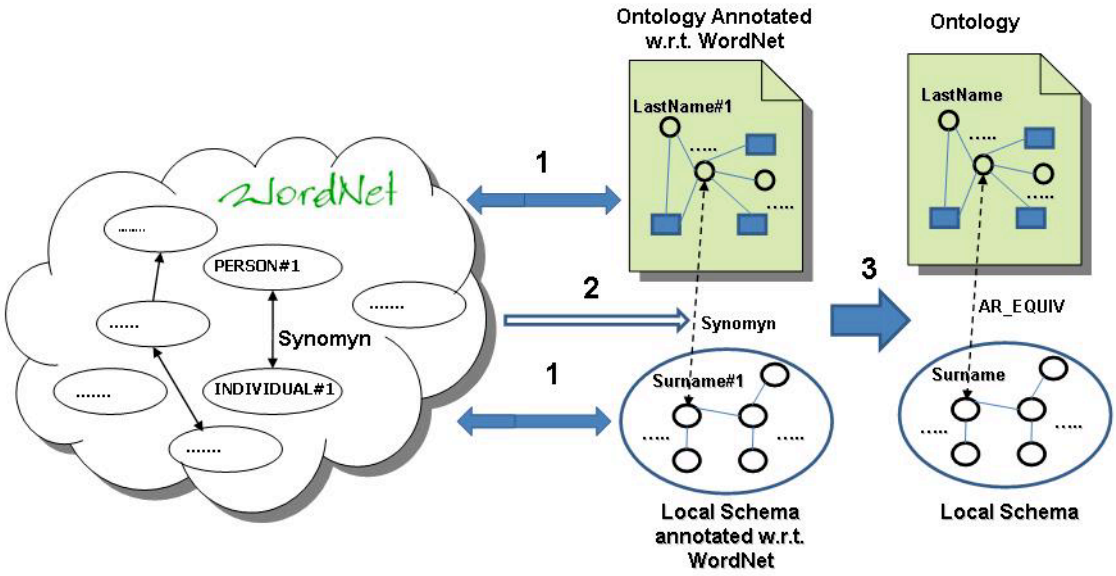

Figure 12 - Performing automatic annotation for Ontology-driven Semantic Mapping

Let us consider the example shows in : during the local source annotation step, the relationships between semantic entities and ontology concepts have to be discovered. Our idea can be summed up in three main steps (each step number is correspondingly represented in figure):

1. *Ontology and local source annotation with respect to WordNet*: both the ontology and the local source, are annotated, with respect to WordNet, by using the Automatic Lexical Annotation method described in [15]: e.g., as shown in **Errore. L'origine riferimento non è stata trovata.**, the semantic entity “Surname” is annotated with the first sense in WN (indicated in Figure with “#1”) for the word “SURNAME” and the ontology concept “LastName” is annotated with the first sense in WN for the word “LASTNAME”;
2. *WordNet semantic relationship discovery*: starting from the previous annotations, a set of WordNet semantic relationships (*synonym* (equivalence), *hypernym* (more general) etc.) is discovered among semantic entities and ontology concepts: e.g., as shown in **Errore. L'origine riferimento non è stata trovata.**, a synonym relationship is discovered between the semantic entity “Surname” and the ontology concept “LastName”.
3. *Local source annotation for Ontology Driven Semantic Mapping*: starting from the set of WordNet semantic relationships previously discovered, a correspondent set of annotation for Ontology-Driven Semantic Mapping can be discovered: e.g., starting from the Word-Net synonym relationship between “Surname” and the “LastName”, the following annotation is established (the annotation unique identifier ID is omitted):

```
(surname, AR_EQUIV, O:LastName)
```

In this way, we can automatically annotate a set of local schema semantic entities with respect to the considered ontology. However, these annotations can be incomplete (because WordNet does not contain many domain dependent words) and require a designer refinement. Even if this preliminary idea needs to be further investigated, it represents a fundamental start point to help the designer during the time consuming task of manual annotation.

Another future work will be the investigation of automatic techniques to discover the relationships among *semantic entities* combining the exploration of multiple and heterogeneous online ontologies with the annotations provided by the WordNet lexical ontology. The use of online ontologies represents an effective way to improve the semantic mapping process. For example, in [18], automatic techniques to discover the relationships between two concepts automatically finding and exploring multiple and heterogeneous online ontologies, have been proposed.

A.4 Conclusions

In this paper, we have described the early effort to obtain an effective Global Schema Generation approach for Ontology-Based Data Integration combining the techniques provided by the MOMIS and the STASIS frameworks. In particular, with the Ontology-driven Semantic Mapping framework we have performed in the Data Integration System the annotation of data sources elements with respect to *generic* ontologies (expressed in OWL). In this way, we have eliminated the MOMIS limitation to use only the lexical ontology WordNet by introducing a *multiple ontology* approach with respect to the actual *single ontology* approach. Moreover, the matching process we have proposed permits to match a schema and ontology, that is a relevant activity in database applications since a number of important database problems have been shown to have improved solutions by using an ontology to provide precise semantics for a database schema [3]. Even if this work needs to be further investigated (as described in Section A.3 Future Work), it represents a fundamental starting point versus a fully automatic Ontology-Based Data Integration System.

References

[Abiteboul1999] Serge Abiteboul, Sophie Cluet, Tova Milo, Pini Mogilevsky, Jérôme Siméon, Sagit Zohar: Tools for Data Translation and Integration. IEEE Data Eng. Bull. 22(1): 3-8 (1999)

[Hai2007] Do Hong Hai: Schema Matching and Mapping-based Data Integration: Architecture, Approaches and Evaluation, VDM Verlag Saarbrücken, Germany 2007

[Kalfoglou2005] Yannis Kalfoglou, W. Marco Schorlemmer: Ontology Mapping: The State of the Art. Semantic Interoperability and Integration 2005

[Zamboulis2008] Lucas Zamboulis, Alexandra Poulovassilis, Jianing Wang: Ontology-Assisted Data Transformation and Integration. ODBIS 2008: 29-36