

LUISA

*Learning Content Management System Using Innovative Semantic Web
Services Architecture*

IST- FP6 - 027149



Deliverable D2.3.2 **Semantic Web Service Infrastructure Implementation – Publication, Discovery, Execution**

Stefan Dietze
Alessio Gugliotta
John Domingue
Tomás Pariente Lobo

Due date of deliverable: 30/05/2008

Actual submission date: 25/06/2008

Start date of the project: 01/03/2006

Duration: 30 Months

Stefan Dietze
Open University, UK

Version 1.0, dated 20/06/2008

Change History

Version	Date	Status	Author (Partner)	Description
0.1	15.06.2008	Draft	Stefan Dietze	First draft proposed.
0.2	17.06.2008	Draft	Alessio Gugliotta	Updated SWS descriptions.
0.3	19.06.2008	Draft	Tomas Pariente Lobo	Updated Negotiation Layer Section
0.4	19.06.2008	Draft	Stefan Dietze	Introduction and conclusion rewritten and minor improvements.
0.5	20.06.2008	Draft	Tomas Pariente Lobo	Consideration of review from Giunti.
0.6	23.06.2008	Draft	Stefan Dietze	Consideration of review remarks from UAH.
0.7	24.06.2008	Draft	Tomas Pariente Lobo	Consideration of review from UAH.
1.0	24.06.2008	Final	Stefan Dietze	Final improvements.

EXECUTIVE SUMMARY

In this deliverable, we define the implementation approach followed to adopt the SWS-based infrastructure (introduced in deliverable D2.1) regarding the requirements described in deliverable D2.2. Starting from an introduction of the requirements, we derive an overall implementation approach. Whereas the conceptual approach foresees the gradual (semantic) abstraction from learning resources based on a set of semantic layers which are aligned to SWS, the architectural approach considers several software components which support the conceptual approach. The remaining sections describe the implementation activities taken to address each of the requirements by following the described implementation approach. While focussing on the additional implementation activities taken with respect to the second LUISA prototype, this deliverable substantially builds on the previous deliverable D2.3.1.

Document Information

IST Project Number	FP6 – 027149	Acronym	LUISA
Full title	Learning Content Management System Using Innovative Semantic Web Services Architecture		
Project URL	http://www.luisa-project.eu/		
Document URL			
EU Project officer	Francesco Barbato		

Deliverable	Number	2.3.2	Title	Semantic Web Service Infrastructure Implementation – Publication, Discovery, Execution
Work package	Number	2	Title	SWS Infrastructure

Date of delivery	Contractual	30/05/2008	Actual	25/06/2008
Status	Version 1.0., dated 20/06/2008		final <input checked="" type="checkbox"/>	
Nature	Report <input checked="" type="checkbox"/> Demonstrator <input type="checkbox"/> Other <input type="checkbox"/>			
Dissemination Level	Public <input checked="" type="checkbox"/> Consortium <input type="checkbox"/>			
Abstract (for dissemination)	This deliverable describes the implementation approach followed to adopt a SWS-based infrastructure for the E-Learning domain.			
Keywords	Learning objects, Learning Object Repositories, Semantic Web Services, Ontologies, WSMO, IRS-III			

Authors (Partner)	Stefan Dietze (OU), Alessio Gugliotta (OU), John Domingue (OU), Tomás Pariente Lobo (ATOS)			
Responsible Author	Stefan Dietze		Email	s.dietze@open.ac.uk
	Partner	OU	Phone	

Project Consortium Information








Partner	Acronym	Contact
Atos Origin S.A.E. (Coordinator)	ATOS 	Nuria de Lama Atos Origin S.A.E. c/ Albasanz 12 E-28037 Madrid, Spain Email: nuria.delama@atosorigin.com Tel.: +34 91 214 9321 Fax: +34 91 754 3252
University of Alcalá de Henares	UAH 	Dr. Miguel-Angel Sicilia Information Research Unit University of Alcalá Ctra. De Barcelona, Km 33.6 E-28871Alcalá de Henares (Madrid), Spain Email: msicilia@uah.es Tel.: +34 91 886 6603 Fax: +34 91 885 6646
University of Uppsala	ULL 	Dr. Ambjorn Naeve University of Uppsala Kyrkogårdsgatan 2 C Uppsala Email: amb@nada.kth.se Fax: +46 184-716-294
Open University	OU 	Dr. John Domingue Knowledge Media Institute, The Open University, Walton Hall, Milton Keynes, MK7 6AA, United Kingdom Email: j.b.domingue@open.ac.uk Tel.: +44 1908 655014 Fax: +44 1908-653-169
University Henri Poincaré	UHP 	Dr. Monique Grandbastien University Henri Poincaré Vandoeuvre les Nancy 54506, PO Box 239, France. Email: monique.grandbastien@loria.fr Fax: +33 383-278-319
Giunti Interactive Labs S.r.l.	GIUNTI 	Fabrizio Giorgini Giunti Interactive Labs S.r.l. Abbazia dell'Annunziata Via Portobello Baia del Silenzio 16039 Sestri Levante (GE), Italy Tel.: +39.0185.42123 Fax: +39.0185.43347
EADS FRANCE – Innovation works	EADS 	Anne Monceaux EADS FRANCE – Innovation works Avenue Didier Daurat - Centreda 1, Toulouse, 31700, France. Email: anne.monceaux@airbus.com Tel.: +33 561-184-725 Fax: +33 561-187-611

TABLE OF CONTENTS

EXECUTIVE SUMMARY	3
TABLE OF CONTENTS	6
1 INTRODUCTION	7
2 SEMANTIC WEB SERVICES REQUIREMENTS FROM THE E-LEARNING DOMAIN	9
3 IMPLEMENTATION APPROACH	11
4 DOMAIN-SPECIFICITY: DERIVATION OF WSMO FOR E-LEARNING SPECIFIC USE CASES	14
4.1 Ontology Stack for the E-Learning Domain.....	14
4.2 Semantic Web Services for E-Learning specific Use Cases.....	15
5 USER INTERACTIVITY THROUGH IMPLEMENTATION OF THE NEGOTIATION LAYER ...	21
5.1 Conceptual approach.....	21
5.2 Architectural Approach.....	22
5.3 Implementation	25
5.3.2 Negotiation	26
5.3.3 Query Resolvers	28
5.3.4 Composition of LOs.....	28
6 CONCLUSIONS	32
REFERENCES	34
7 APPENDIX	37
7.1 SWS Descriptions aimed at retrieving (EADS) LOM given a specific LOM field	37
7.2 SWS Descriptions aimed at retrieving (UHP) LOM by Topic	45
7.3 SWS Descriptions aimed at retrieving both UHP and EADS LOM by Competency.....	47
7.4 Negotiation Layer WSDL	50
7.5 Query Resolver Java Interface	70

1 INTRODUCTION

The increasing availability of learning resources on the Web raises the need of discovering and delivering to the learner the most appropriate learning resource, given the specific learner needs and his/her actual learning situation. The latter is defined by e. g. the used technical environment or specific learner characteristics - such as his/her native language - and constitutes the actual context which has to be addressed appropriately.

The current state of the art in E-Learning is mainly represented by approaches based on software systems – e. g. learning content management systems (LCMS) – which provide a learner with composite learning contents – the so called Learning Objects (LO) which are composed manually at design-time. To enable interoperability between several platforms and applications, metadata standards, such as IEEE LOM [10], ADL SCORM [1] or IMS Learning Design (IMS LD) [18], are available. These aim to provide a common specification for describing complex learning objects as well as the learning process to be followed by the learner. Due to the design-time allocation of learning resources – learning assets are selected and composed manually during the design-process of composed LOs - the actual context of the learning process cannot be considered appropriately and a learning object cannot adapt dynamically to the specific context or learner needs. Therefore, reusability across contexts, metadata standards and platforms is limited and increasing costs for allocating resources have to be taken into account. Moreover, current approaches do not allow the integration of several distributed LO repositories, even though this is a major requirement in order to obtain the most appropriate resource for a given need and context.

The use of services instead of data addresses these issues partially. For instance, by allocating Web services instead of static resources, the resulting learning process will be the sequence of simple or composed functionalities that dynamically target the learning objectives by obtaining distributed resources at runtime. However, since Web services are deployed using purely syntactic technologies such as SOAP [32], WSDL [37], and UDDI [35], which do not provide information about the semantic meaning of the service functionalities, utilized data or usage constraints, services cannot be discovered, composed and invoked automatically.

Summarizing the current lack, existing metadata standards enable context-adaptation based on conditional selection of a pre-defined set of learning data, whereas the usage of Web services instead of static data introduces a dynamic discovery and delivery of data out of specific repositories. But both approaches do not support context-adaptation and integration of distributed heterogeneous learning resources based on the dynamic allocation and orchestration of services utilizing comprehensive semantics and consequently, the context-appropriate run-time allocation of services and data out of a virtually unlimited number of distributed repositories.

To address these issues and requirements – a detailed elaboration can be found in D2.2 - we propose a Semantic Web Service (SWS) based application framework and describe the required implementation activities in this document.

As introduced in the deliverable D2.1, we base our approach on the Web Service Modelling Ontology (WSMO) [38] and use IRS-III [3] as reference SWS-environment. However, WSMO and IRS-III are general purpose means. In this document, we focus on the implementation activities necessary to address the specific requirements which apply to the adaptation of WSMO and IRS-III for the domain of E-Learning (D2.2). Note that this deliverable builds to a large extent the previous deliverable D2.3.1 [8] and focuses on the additional implementation activities necessary to provide the second LUISA prototype. Hence, throughout the remaining document we continuously refer to D2.3.1.

The rest of the paper is organized as follows. Section 2 summarizes the requirements to the adaptation of WSMO for E-Learning, whereas Section 3 introduces the overall approach to tackle these requirements. Section 4 proposes a set of ontologies and SWS descriptions which gradually derive WSMO for E-Learning-specific use cases. The implementation approach followed to enable user negotiations during SWS orchestration is described in Section 5. Finally, Section 6 reports our conclusions.

2 SEMANTIC WEB SERVICES REQUIREMENTS FROM THE E-LEARNING DOMAIN

SWS technologies are general purpose technologies which enable the application to distinct domains of usage – for instance E-Learning or Business Processes. On the one hand, this domain-independence has to be perceived as a crucial important advantage of SWS technology, since it allows to benefit from SWS within every domain. However, this characteristic also leads to the demand of adopting SWS for each distinct domain. Thus, an important part of the work within LUISA WP 2 is concerned with the adaptation and extension of SWS technology – namely WSMO and IRS III - to enable the domain specific usage within the domain of E-Learning.

The general purpose scope of WSMO and IRS III and the issues described in Section 2 lead us to the following requirements which have to be considered when adopting SWS technology for the E-Learning domain:

- R1. **Integration:** enabling integration and furthermore, a run-time allocation of distributed E-Learning resources. This requirement directly is derived from the use cases (WP6 and WP7), where the integration of distinct institutional repositories is required. The integration of distributed heterogeneous resources has to consider the integration of learning-related services as well as data. This requirement is the more abstract and, therefore, its achievement is based on the achievement of the following more specific requirements.
- R2. **Domain-specificity:** introducing E-Learning specific descriptions of SWS and learning resources. This requirement directly relates to the need for representing distinct institutional vocabularies of the used E-Learning concepts, such as competencies and LOs, as proposed in WP6 and Wp7. It is important to support the development of SWS based E-Learning applications by enabling domain-specific descriptions of available SWS which consequently base a SWS description on learning-related concepts such as competencies.
- R3. **Negotiation:** identifying the actual learning situation through an active negotiation with the learner. To enable most fine-grain situation-awareness, it is necessary to enable the negotiation with the end-user to gradually refine his/her actual need as well as his/her actual situation. For instance, as proposed by WP6 and WP7, a user has to be able to describe his/her current learning need and to gradually refine his needs based on the responses of the learning environment regarding the available resources.
- R4. **Context-Awareness:** enabling resource allocation according to the specific situation (context). It is important to consider the actual learning situation in which the learner is acting, e.g. defined by the preferences of the learner, his particular native language or technical environment when allocating appropriate learning resources.
- R5. **Fuzzy Matchmaking of Context Parameters:** To enable context-awareness (R4) usually a possibly infinite set of real-world context

characteristics, has to be matched to a (semantically) predefined set of parameter descriptions. This is of particular interest, since usually not any real-world characteristic, such as the competencies of a learner, completely equals another. Hence, describing the complex notion of a particular E-Learning situation is a costly task and it might not be feasible, to represent all potential context parameters through symbolic representation approach. Therefore, we claim that one-to-one matchmaking is not feasible in many cases and hence, fuzzy matchmaking methods are required to match between real-world situation characteristics and predefined parameter representations.

- R6. **Standards-interoperability:** keeping interoperability with existing E-Learning standards. Since E-Learning applications often make use of specific E-Learning metadata standards – e. g. IMS LD, ADL SCORM – the compliancy with such standards as well as the interoperability between them has to be perceived as a crucial important need.

All of these requirements are addressed by extending the current SWS technologies – WSMO and IRS III. Please note, that requirements R1-R3 are directly derived by considering the use cases provided in WP6 and WP7 and are addressed by the currently integrated prototype of the LUISA framework. In contrast, R4-R5 are considered as relevant for the application of SWS to the E-Learning domain and are addressed through further implementation activities. This led to further semantic and architectural facilities which are executable within a separate proof-of-concept prototype application but not the overall LUISA prototype application. However, it is intended to integrate the components from both prototype applications into a common application framework within future work.

The implementation activities to consider these requirements are elaborated in the following sections.

3 IMPLEMENTATION APPROACH

This section describes our overall approach to consider R1-R5 when adopting IRS-III.

We consider the automatic allocation of resources at runtime of a learning process. Learning processes are described semantically as a composition of user objectives (goals) within a specific learning context. The learning process models abstract from specific resources – whether data or services - and metadata standards. Based on the semantic descriptions of available resources, the most appropriate resource is selected automatically to achieve a certain learning goal within the actual learning (runtime) context. This vision enables a highly dynamic adaptation to different learning contexts and learner needs. Moreover, using adequate mappings, our standard-independent process models can be translated into existing metadata standards in order to enable reuse within existing standard-compliant runtime environments.

In this way, we consequently reduce the efforts of creating learning process models: one unique learning process model can adapt dynamically to different process contexts and can be translated into different process metadata standards.

Our approach is fundamentally based on realizing the following principles:

P1. Contextualization of learning process models.

P2. Learning resource abstraction.

To support these principles, we introduce a layered approach to achieve a gradual abstraction and finally, a gradual mapping between resources – data and services – and process metadata (Figure 1). In the figure below, dashed lines represent user interfaces which are intended within the framework, whereas drawn through lines represent interfaces which are already implemented within the current application framework:

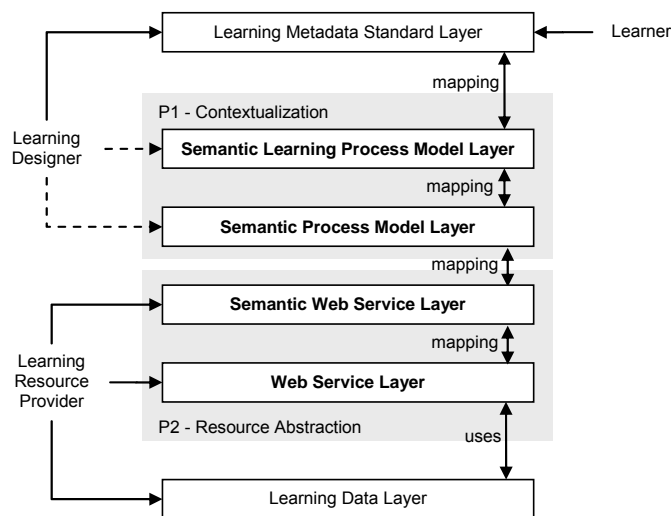


Figure 1. Conceptual framework to gradually abstract from learning process resources and contextualize learning process model

P1. Contextualization of Learning Process Models. A first layer concerned with the semantic contextualization of current learning process metadata standards is the *Semantic Learning Process Model Layer*. It allows the comprehensive description of contextualized processes within the domain of E-Learning in terms of higher level domain concepts - e. g. learning goals, learners or any other kind of context parameter. This layer is mapped to semantic representations of current learning metadata standards in order to enable the interoperability between different standards and furthermore, the automatic translation between different metadata standards.

To achieve a further abstraction from domain specific process models – whether it is e. g. a learning process, a business process or a communication process – we consider an upper level process model layer – *Semantic Process Model Layer*. This layer enables the description of contextualized domain-independent processes and introduces the high-level concepts – e. g. process goals, roles or process parameters – which can be specialized for different process domains. Thus, this layer enables the alignment of different domain-specific Process Model Layers and therefore, a mapping between them - for instance the mapping between learning objectives and business objectives.

P2. Learning Resource Abstraction. To abstract from existing learning data and content we consider a *Web Service Layer* which exposes functionalities appropriate to fulfill specific learning objectives. These functionalities range from, for instance, querying learning data repositories to filtering of data or to compute competency gaps. This abstraction from learning process data enables a dynamic discovery of appropriate data to suit a specific context and objective. Services exposed at this layer may make use of semantic descriptions of available learning data to accomplish their functionalities.

In order to abstract from these functionalities (Web services), we introduce an additional layer – the *Semantic Web Service Layer*. This layer enables the dynamic selection, composition and invocation of appropriate Web services for a specific learning objective within a particular learning context. This is achieved on the basis of formal semantic, declarative descriptions of the capabilities of available services which enable the dynamic matching of service capabilities to specific user goals. Due to the semantic abstraction from learning resources, additional distributed resources can be integrated into the application framework by simply adding semantic resource descriptions (*Semantic Web Service Layer*) following a common SWS standard, such as WSMO [38].

Based on mappings, upper level layers can utilize information at lower level layers. It is important to note, that we explicitly consider mappings not only between multiple semantic layers but also within a specific semantic layer.

The following sections provide details about the implementation approach followed to support the conceptual framework described in this section aimed at addressing R1-R6. This procedure particularly includes the following activities:

- A1. Provision of architectural components: In a first step, all architectural components of the SWS Layer have been implemented. This particularly includes the set up and configuration of the SWS Broker IRS-III, the

Transparency Layer which provides a transparent API on top of IRS-III to the Negotiation Layer (Section 5).

- A2. Population of Data Layer: To support the Data Layer, semantic metadata describing learning resources and learner profiles have been provided and imported into Learning Object Metadata Repositories (LOMR). Since these activities are part of WP4, WP6 and WP7, these are not described in this document.
- A3. Population of Web Service Layer: In order to populate the Web Service Layer, several distributed Web Services have been set up. These provide functionalities such as the retrieval of Learning Object Metadata (LOM) record sets, the retrieval of learner profiles or the computation of competency gaps. Web services have been exposed by utilizing different implementation approaches, ranging from standard WSDL-based [37] services to LISP functions exposed as SWS.
- A4. Semantic Representation at SWS Layer: Web services have been described semantically as SWS based on WSMO. This usually considers (i) the description of goal representing the outcome of a Web service from a user perspective, (ii) the semantic representation of the service capabilities, and (iii) the introduction of mediation facilities which deal with semantic and syntactic interoperability issues between distinct service providers and consumers.

4 DOMAIN-SPECIFICITY: DERIVATION OF WSMO FOR E-LEARNING SPECIFIC USE CASES

To support the semantic layers described in the previous section, we introduce an ontology stack aimed at supporting the aforementioned semantic layers. The newly introduced ontologies together with the SWS descriptions supporting the use cases within LUISA build the semantic foundations to reason on E-Learning specific scenarios and are build and hosted within IRS-III. Even though parts are specifically build for the use cases provided in WP6 and WP7, parts are reusable across further scenarios and enable the deployment of LUISA technologies in further application settings.

4.1 Ontology Stack for the E-Learning Domain

The stack illustrates how each semantic layer is populated through the use of ontologies which gradually bridge between learning resources and SWS. The following Figure 2 depicts the ontology stack which has been provided.

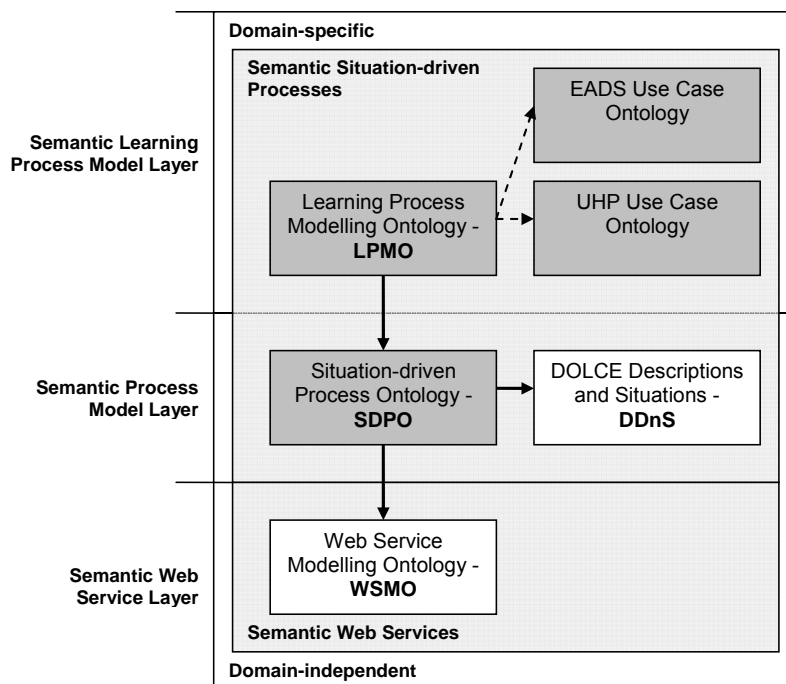


Figure 2. Utilized stack of ontologies.

The grey-coloured boxes in Figure 2 represent ontologies which have been provided within LUISA, whereas the white boxes represent established ontologies which have been utilized and derived for the purposes of LUISA. The Learning Process Modelling Ontology (LPMO) implements the Semantic Learning Process Model Layer and is aimed at modelling a learning process from a general point of view - independent from any supported platform or learning technology standard. For each of the use cases, designated ontologies have been created - EADS Use Case Ontology and UHP Use Case Ontology – which provide the semantic vocabulary representation to annotate learning resources within each of the use case scenarios. To enable mappings between

concepts across both use cases, both ontologies are partially aligned to LPMO. As detailed in the following sections, these mappings have been implemented in the second prototype, in order to provide an added value SWS, which queries for learning objects across both use case domains by utilising semantic mappings. Furthermore, LPMO is mapped to ontological representations of E-Learning metadata standards to support interoperability (see [8]).

The domain-independent Situation-Driven Process Ontology (SDPO) abstracts from the process domain and implements the Semantic Process Model Layer. SDPO will enable the description of contextualized processes independent from its specific purpose and allows mappings to domain-specific process ontologies such as the LPMO. In order to enable a high level of interoperability of our ontologies, the SDPO as well as the LPMO are aligned to the DOLCE foundational ontology [17]. In particular, context descriptions are based on the Descriptions and Situations module (DDnS) [16] of Dolce. Furthermore, the SDPO is mapped to the WSMO standard. Therefore, these ontologies realise a gradual derivation of WSMO entities for the E-Learning domain. It has to be highlighted, that our ontological architecture explicitly considers mappings not only between several semantic layers but also within a specific semantic layer. This enables for example the mapping of our LPMO concepts to other existing semantic descriptions of learning related concepts.

4.2 Semantic Web Services for E-Learning specific Use Cases

To support the use case scenarios provided by EADS and UHP in WP6 and WP7, a set of SWS descriptions has been provided. The created SWS descriptions either semantically represent the Web services related to their use case scenarios or opportunely combine available SWS descriptions to provide added value services. In particular, we distinguish three sets of SWS descriptions for each of the partner:

- S1. Profile-related SWS
- S2. Learning Object-related SWS
- S3. Extended Learning Object-related SWS

The first one refers to Web services that operate on top of profile databases from each partner and is aimed at retrieving and manipulating learner profiles. The second one refers to Web services that operate on top of the LOMR (WP4) and aims at providing access to semantic annotations of Learning Objects. The third one extends the previous set with additional SWS descriptions, which refer to the SWS descriptions available in S2 and either represent new Web services on top of the LOMR or implement completely new functionalities. Please note that S1 and S2 have been already introduced in the first version of this deliverable (D2.3.1). Therefore, just a summary is reported below, while the focus of this version of the deliverable is on S3.

Source code listings for SWS descriptions within S3 are enclosed in the Appendix of this deliverable, while the ones within S1 and S2 have been enclosed in the Appendix of D2.3.1 [8]. Please note, that in the following we refer to a numbering of goals which is equivalent for both use cases, the UHP

one and the EADS one. For instance, there does exist a goal G.1 in the domain of the UHP use case and a distinct G.1 in the domain of the EADS case.

S1. Profile-related SWS

A general purpose SWS goal (*G.1, Get UHP Profile*) has been provided to expose different kind of learner profiles for a given user-ID. G1 is invoked through the Negotiation Layer (Section 5) each time a learner profile is requested. The goal is associated through a Wg-Mediator with three different SWS. Based on semantic capability descriptions of each SWS, IRS-III automatically selects the most appropriate service for a given goal invocation.

Figure 3 depicts the organization of the SWS descriptions for the UHP use case aimed at retrieving learner profiles. The SWS representations for the EADS use case are quite similar and follow the same structure and approach.

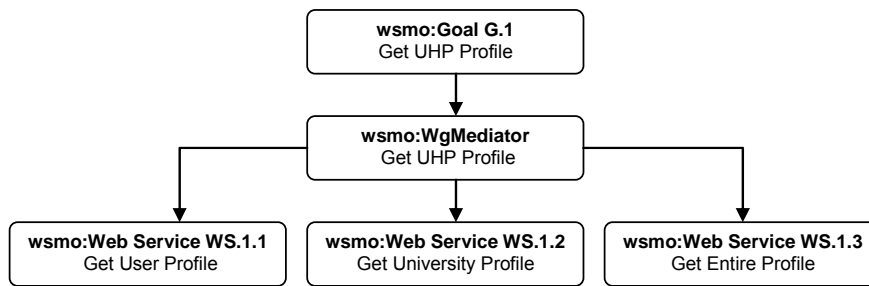


Figure 3. WSMO-based SWS selection to retrieve UHP profiles.

S2. Learning Object-related SWS

In contrast to the previous set, several distinct services have been provided to expose Learning Object Metadata (LOM) out of the use case specific LOMRs.

For the retrieval of UHP LOM, three SWS goals have been provided: the first one (G.3) aims at retrieving LO references for a given competency; the second one (G.4) aims at retrieving LOM for a set of given LO references; and the third one (G.2) defines a meta-goal (i.e. not grounded to an actual Web service), which defines the orchestration of the other two, as well as the mediation between the outputs of the first and inputs of the second. Figure 6 depicts the relation between G2, G3 and G4.

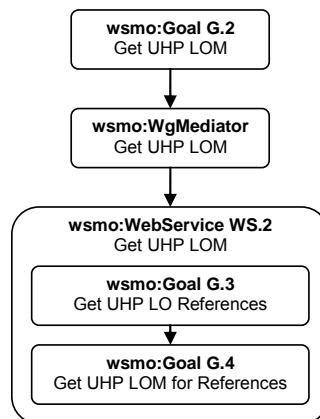


Figure 4. WSMO-based SWS orchestration of two SWS to retrieve appropriate LOM records for a given competency.

Notice that the SWS representations for the EADS use case are quite similar and follow the same structure and approach.

S3. Extended Learning Object-related SWS

Distinctive to the SWS descriptions of the previous two sets, the SWS descriptions of S3 make use of existing SWS descriptions and propose more complex structures, exploiting together many benefits of using SWS - such as easy integration and reuse, runtime selection of services, orchestration and mediation among distinct representations and implementations.

In this set, we mainly introduced the following three new functionalities:

- F1.get LOM by specifying one of the LOM fields (keyword, code, function, field, domain or speciality) in the EADS domain;
- F2.get LOM by specifying the topic in the UHP domain;
- F3.get LOM by competency (as in S2) by querying both the UHP and EADS domain (i.e. realizing the mapping between the two domains).

In order to implement the first functionality (F1), we followed the structure depicted in Figure 5.

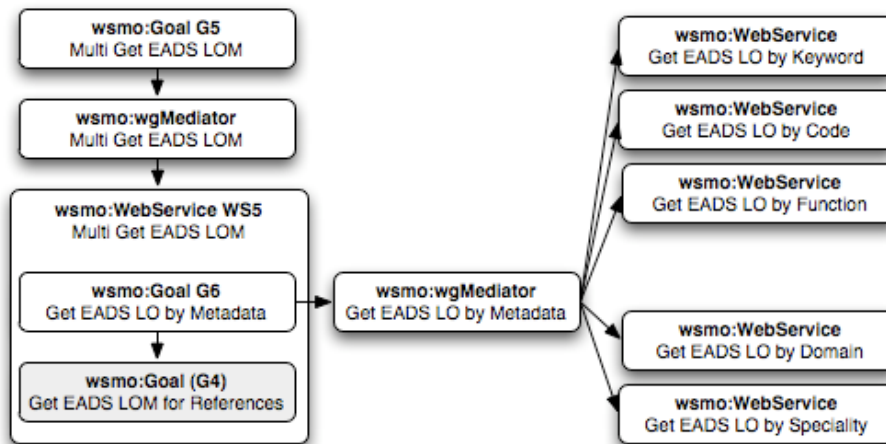


Figure 5. Structure of WSMO-based SWS descriptions to retrieve appropriate EADS LOM records for a given LOM field. The grey box represents a definition already available in S2.

The proposed structure reuses a goal defined within the S2 set (G4 for EADS use case) and combines orchestration of goals and runtime selection of the most appropriate service. In fact, the main goal (G5) is decomposed into two sub-goals within the linked Web service description.

In turn, the first sub-goal (G6) is linked to a set of Semantic Web service descriptions. Each Web service description grounds to a specific operation of a Web service operating on top of the EADS LOMR. The most appropriate operation is selected and invoked at runtime by IRS on the basis of the stated Web service assumptions and the actual inputs IRS provided when invoking the goal G5. The following OCML code shows an example of Web service capability description, comprehensive of the respective assumption expression:

```

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-WS-CAPABILITY (CAPABILITY) ?CAPABILITY
((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
(HAS-ASSUMPTION :VALUE
(KAPPA (?WEB-SERVICE)
(= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
"keyword")))))
  
```

The assumption expression states that the input value of the input role *has-search* of the goal G5 should be equal to the string "keyword", since the underlying WS is querying the LOMR based on a set of keywords. In this way, a unique goal (G5) has been mapped to many possible ways of accomplish such a goal (by keyword, code, function etc.).

The second sub-goal of the orchestration (G4 for the EADS use case) aims at retrieving metadata descriptions of a LO, given a specific LO reference. This subgoal has been elaborated in the previous deliverable D2.3.1 [8]. The specific list of metadata fields to retrieve is specified as input of the meta-goal (G5) invocations, while the list of LO references is the output of the previous sub-goal

G6. Notice that reusing an existing goal within an orchestration is just a matter of reporting its name within a sequence definition (unless specific mediation facilities are required).

For the second functionality (F2), we followed a structure similar to the previous one. Particularly, the second goal of the orchestration is also a goal already available in S2; i.e. the goal G4 in the UHP domain. The only difference in the structure is that the first goal of the orchestration is currently linked to a unique Web service description. However, additional Web service descriptions can be easily added in future by linking them to the existing wg-mediator and stating the appropriate assumption expressions.

Differently to all the previous functionalities, the third functionality (F3) is addressed by implementing a meta-goal that crosses the two use case domains by introducing a (service-based) mediation facility. Figure 6 depicts the respective structure.

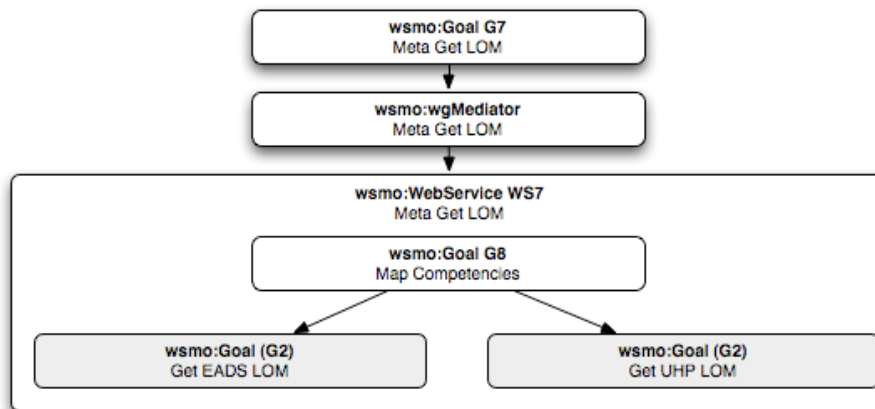


Figure 6. Structure of WSMO-based SWS descriptions to query both the UHP and EADS domains by competency.

The meta-goal (G7) can be invoked by both UHP and EADS users, presenting the required competency in the respective format. Notice that the LOMR of one domain is not aware of the competencies represented in the LOMRs of other domains. The mappings are represented at the LPMO level, exploiting the ontology stack introduced in Section 4.1 (Figure 4), and accessed by achieving the first goal of the represented orchestration (G8). To put it in other words, two distinct ontologies – one for the UHP case and one for the EADS case - had been provided, representing the terminology utilised within each of the respective LOMRs. These ontologies had been aligned to upper-level concepts described within LPMO, in order to implicitly provide a mapping across both use case vocabularies. These mappings particularly include mappings between distinct competency representations, which are used to enable queries across both available LOMRs.

Reasoning on the provided mappings is accomplished by a Web service that (i) receives as input a list of competencies - either in a UHP or EADS form -, (ii) reasons on the available LPMO descriptions and (iii) returns two lists with the initial and the mapped competencies, respectively. The accomplishments of G8 triggers the execution (in sequence) of G4 in the UHP and EADS domain.

Notice that these two meta-goals are already provided for S2 but are incorporated into the proposed orchestration here. The respective results are then combined into a unique list and returned to the negotiation layer.

5 USER INTERACTIVITY THROUGH IMPLEMENTATION OF THE NEGOTIATION LAYER

The Negotiation Layer is an intermediary layer between the GUI and the Semantic Web Services environment. The main objective of this layer is the identification and ranking of the best learning material through an active negotiation with the learner, posing the most suitable query to retrieve the learning results. One of the requirements is considering the user profile (mostly preferences and competency gap) in order to gradually refine the learner needs according to their current situation. This negotiation involves taking into account the learner context, and the definition and implementation of a set of rules that govern the fulfilment of the user goals according to specific organizational needs.

By abstracting from the actual implementation of the SWS environment through IRS-III, the Negotiation Layer provides implementation independent access to learning resources. Even though the Negotiation Layer takes advantage of the dynamic service discovery capabilities of the underlying SWS environment IRS-III it provides the necessary abstractions which enable the integration of distinct underlying technologies, for instance other SWS environments or even traditional Web services-based solutions.

5.1 Conceptual approach

As it was explained in deliverable D2.2, our approach for negotiation is based on realizing the following steps:

- S1. Infrastructure for negotiation and goal invocation.
- S2. Application of organizational specific e-Learning rules.
- S3. Composition of Learning Objects.

To realise these steps, we introduce a conceptual view of the negotiation that tackles these aspects (Figure 7):

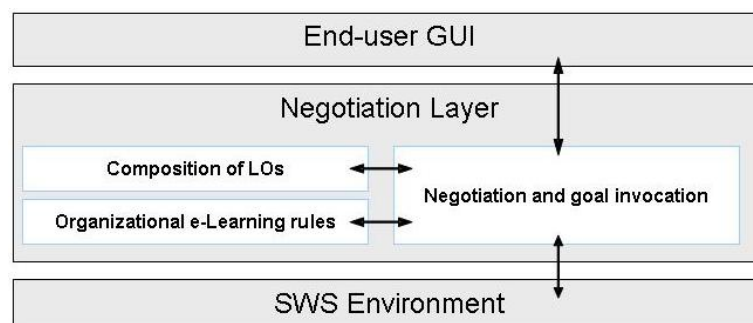


Figure 7. Conceptual Framework for Negotiation

S1. Infrastructure for negotiation and goal invocation. The *Negotiation* component is introduced in order to abstract the learner queries from the Semantic Web Service environment. In this sense, the Negotiation Layer is in

charge of exposing functionalities that decouple the end-user needs to the underlying infrastructure and provide negotiation functionalities. These functionalities range from providing to the end-user layer the learner profile, selecting the WSMO goals from the user needs, or managing the responses from the SWS environment using the goal invocation mechanisms provided by the Transparency Layer.

S2. Application of organizational specific e-Learning rules. The *Organizational e-Learning Rules* layer is where the queries of the learners are matched against a set of organisational well defined rules for negotiation. These rules will cover aspects such as, for instance, what to do when no results have been returned that fulfils the learner needs. Relaxation of the learning objectives following a set of rules defined for the organization in a given scenario is among the functionalities offered by this layer. It is worth noticing that organizational rules may vary depending on the context of the learner, even in the same organization. For instance, in the case of an industrial implementation, the system may decide based on the user or company preferences, or in any other contextual information, between choosing very strict rules, not allowing the system to select Learning Objects that not fulfil its complete learning gap, or more relaxed rules, in which case the system will ask for LOs that fulfil part of the gap detected for the learner. Apart from these rules, the system must allow to implement rules for ranking the results.

The business and ranking rules are then specific to each organization. The system must allow binding the most adequate set of rules.

S3. Composition of Learning Objects. The negotiation layer must allow the composition of Learning Objects according to a set of rules. Learning object composition is the process of combining several LOs into a higher level instructional unit that fulfils specific learning goals under concrete contextual circumstances. Composition requires selection as a constituent process. Nonetheless, composition imposes additional constraints coming from the fact that the LOs retrieved from different sub-goals could not be joined together. For instance, due to incompatible style designs or even to inconsistency in the pedagogical approach or level of description.

There could be different composition strategies. For the second prototype of UHP, a functionality that allows tutors to propose and create learning object composition based on pedagogical rules has been developed. The architecture should be open enough to allow the addition of multiple “instances” of composition components that could be manually or automatically selected.

5.2 Architectural Approach

The Negotiation Layer provides the relationship between the end-users of the LUISA framework and the Semantic Web Service environment. The main objectives of this layer are the following:

- To manage the interaction between the end-user layer and the Semantic Web Service Infrastructure.
- Dealing with the negotiation of the provided results to the end-users.

- To implement the infrastructure to easily plug in the rules of a given organization. These rules are based on specific scenarios, so the infrastructure has to deal with the possibility of adding and selecting between multiple rule-based components tailored to fulfill use needs of the scenario (Query Resolvers).
- Allowing the implementation of different composition strategies in a unique learning package. As in the previous case, the infrastructure has to allow an easy plug in of different strategies (Composers).

Some of the components of the Negotiation Layer are closely related with the implementation of the use case requirements. The following figure shows a general vision of the LUISA architecture giving specific, although not complete, details of the services and architectural components developed for the use cases:

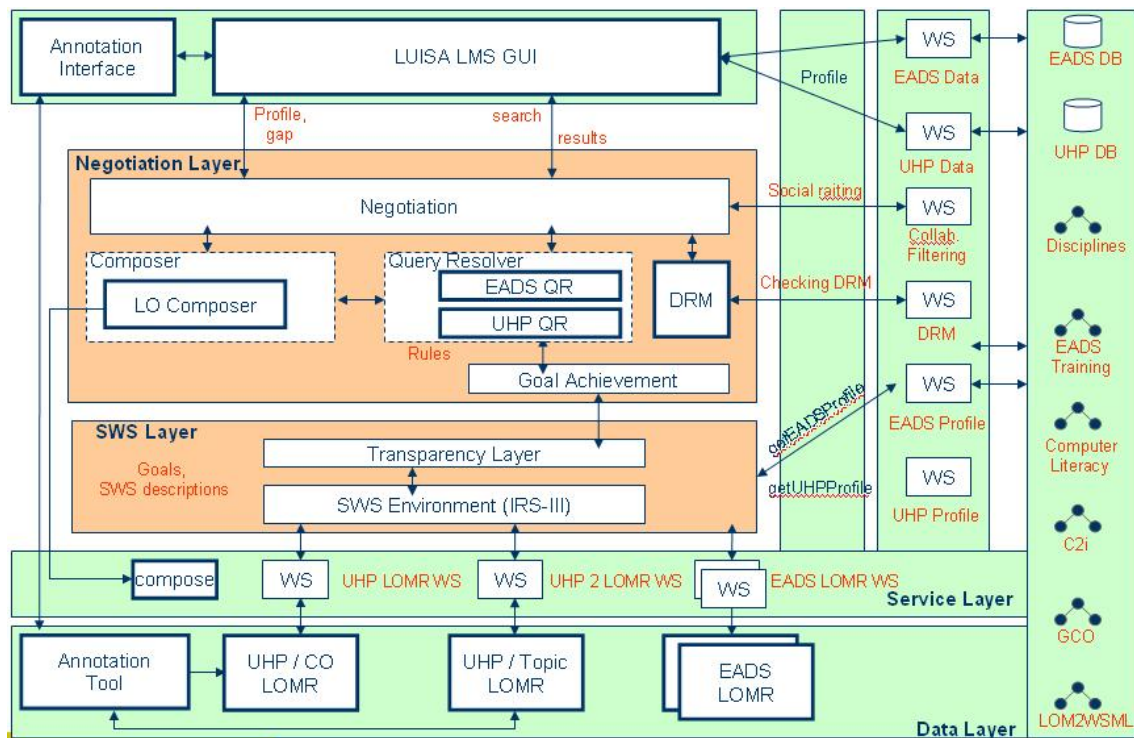


Figure 8. General LUISA Architecture showing use cases-related services

From this picture it is possible to derive some of the main implementation aspects of the Negotiation Layer:

- The Negotiation Layer offers Web Service interfaces to communicate with the User Layer and interacts with the SWS Layer using the goal invocation methods provided by the Transparency Layer.
- The Negotiation Layer implements use cases-related components, such as the Query Resolvers for EADS and UHP.

- The Negotiation Layer interacts with some extra services, such as the Digital Rights Management (DRM) module, and a collaborative filtering engine (CoFe¹).

The following figure shows a finer grained detail of the Negotiation Layer architectural components:

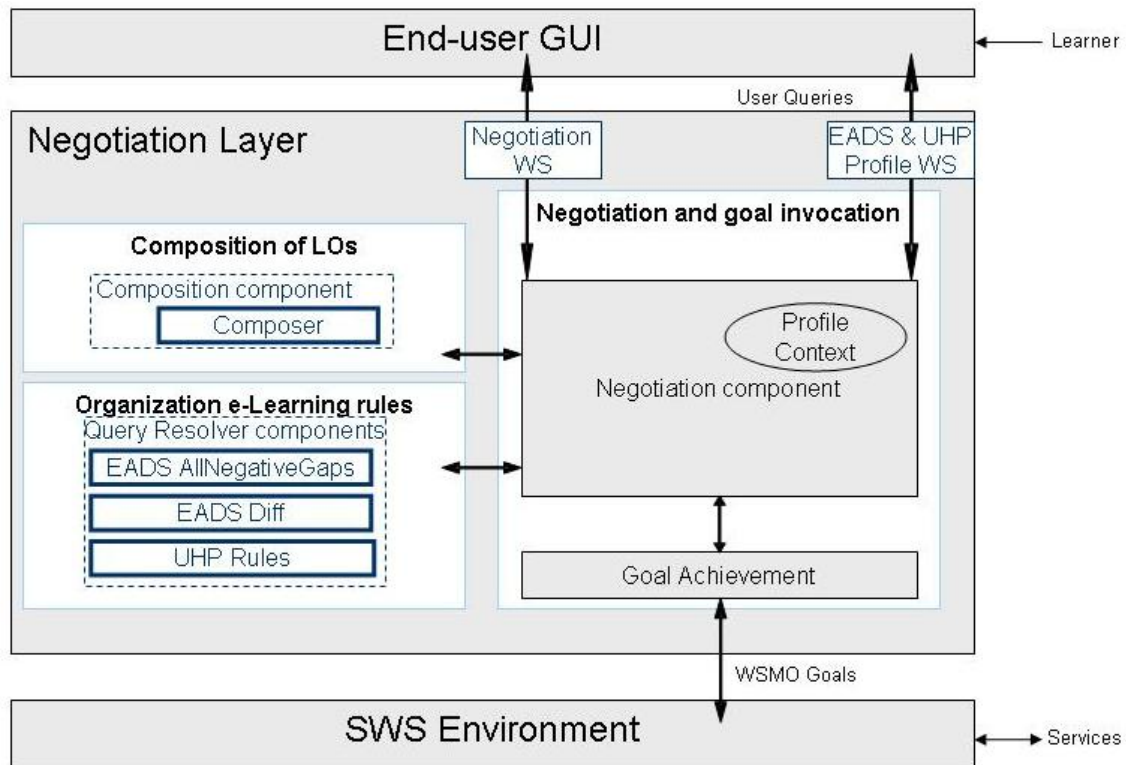


Figure 9. Negotiation Layer Architecture

The process of finding the appropriate learning objects starts with the raise of some kind of learning need submitted by the user of the LUISA system by the GUI. This need can be expressed in simple terms as goals like “learner L from a university requires learning about competency C”, but these goals could also be expressed as complex learning needs. Such complex descriptions may include requirements on the profile, previous competencies of the learner, discipline the learner wants to learn about, constraints of the technical platform in which the learning objects will be delivered, limitations on cost, duration or calendar of the resulting on-line learning design. These learning goals are then decomposable, in the sense that an overall goal can be broken up into simpler sub-goals that may be subject to independent inquiry.

In practical terms, the architecture tries to decouple the GUI from the rest of the layers using a SOA approach. The GUI sends to the Negotiation Layer a simple query that contains the user request. The process of learning is then

¹ CoFe a collaborative filtering engine: <http://eecs.oregonstate.edu/iis/CoFE/>

dependable on the learning goal and the context. Then, the Negotiation Layer must take into account both the decomposition of learning goals into WSMO goals and the context (user profile). Once the WSMO goals are selected, the discovery of the candidate learning objects that fulfill the goals must be triggered, connecting with the Semantic Web Service environment.

Once the candidate LOs have been returned by the SWS environment, the negotiation phase continues by taking into account the requirements from the case studies. The case study rules are applied to refine the list of LOs applying the user context. This issue may lead to perform new queries in order to retrieve a new set of LOs that fit better the original learning needs, by relaxing constraints or applying additional rules to changing the query parameters. As a matter of example, in the UHP case study the following rules are applied:

- The original competencies send by the GUI are part of the first goal invocation.
- If no LO is retrieved from the fist invocation, the original set of competencies are de-composed in sub-competencies from the UHP General Competency Ontology (GCO).
- In any case, the system performs a new goal invocation using the “about” relationship of the GCO between the original competencies and the topics in UHP Computer Literacy ontology.
- The complete set of LO are then ranked according to the user preferences.

The Negotiation Layer also is in charge of the coordination with the DRM and Collaborative Filtering architectural elements. The DRM integration is deeply discussed in D5.3. Regarding the collaborative filtering, the Negotiation Layer integrates an experiment of using social ratings from a collaborative filtering engine (CoFe) that affect the ranking of the Learning Objects for the UHP case study. More details on this experiment can be found in D7.3.2 [25].

5.3 Implementation

Regarding the implementation of the Negotiation Layer components and their interfaces to other architectural layers, some issues were considered in order to reach the following implementation principles:

- S1. The variability of the learning goals and its inputs and outputs associated can be very high (Learning design based on competencies, competency gaps, field of study, etc.). In practical terms, that means that the input and output parameters, or even the interfaces to be provided by the Negotiation Layer to the end-user Layer should be flexible enough to cater with this variability.
- S2. The End-user Layer can be implemented using different technologies (Java, .NET, etc.). This means that the Negotiation Layer must be accessible by multiple technologies.
- S3. The Negotiation Layer is implemented in Java.

Taking into account these issues, the Negotiation Layer software is deployed as a Web Service offering the methods needed by the LUISA case studies to the End-user Layer. This Web Service interface could be further customized to allow future case study requirements.

The core Negotiation component implementation does not depend on the case study implementation, delegating this dependency to the appropriate Query Resolver and Composition components.

The Negotiation Layer is a server-side J2EE application. It is deployed easily by using standard J2EE technology, using a WAR file that includes all necessary classes, libraries and Web service description. The infrastructure needed to deploy the current version of the Negotiation Layer is the following:

Apache Tomcat (web container where the WAR file is deployed)

MySQL (EADS and UHP profile databases)

The Negotiation Layer Web service use Apache Axis 1.4 libraries included in the WAR file (no installation of Axis is needed in the server). This infrastructure applies only to the current setting of the implementation. For instance, as a J2EE application, the software could be configured in any other J2EE compliant application server or use any other SQL-compliant database.

More details in the implementation and deployment of the Negotiation Layer and the specific functionality for the two LUISA case studies can be found in deliverables D6.3.2 [25] and D.7.3.2 [26] .

5.3.2 Negotiation

The Negotiation component allows the management of the communication between end-user layer and the rest of the LUISA framework. The negotiation involves collecting data from the end-user and implies verifying the suitability of the user profile to specific Learning Objects, and assuring the persistence of the session with the GUI.

After receiving the results of the invocation of the SWS environment, these results are gathered and digested by the appropriate Query Resolver that implements the rules to which the invocation are bound. If the results can be combined automatically in new learning packages, the Negotiation component invokes the appropriate Composition component to create this package. The interaction with the end-user is always managed through the Negotiation component.

The negotiation component provides functionalities such as *goal finding and decomposition* -a given query from the user (learning objective) involves finding the appropriate WSMO goal or decomposing it into sub-goals (in case the goal is decomposable)-; *goal Achievement* -component in charge of the invocation of the Semantic Web Service environment using the WSMO goals, which allows that the results to the execution will be digested by the appropriate Query Resolver to apply the organizational rules-; and *session persistence* in order to maintain a conversational session with the end user.

The Negotiation Layer Web Service offers the following operation set to the GUI:

wsdl:portType		NegotiationLayerWS	
wsdl:operation (22)		name	parameter0...
1	getProfileEADS	userId	
2	getExplanation	correlationIdIn	
3	deleteProfileUHP	userId correlationIdIn	
4	deleteProfileEADS	userId correlationIdIn	
5	getProfileUHP	userId	
6	savePreferencesEADS	correlationIdIn preferences	
7	savePreferencesUHP	correlationIdIn preferences	
8	saveProfileEADS	userId profile	
9	saveProfileUHP	userId profile	
10	saveIndustryPreferencesEADS	correlationIdIn preferences	
11	saveUniversityPreferencesUHP	correlationIdIn preferences	
12	getCOsByProfession	correlationIdIn profession	
13	getLOsForCompetenciesOtherProfessionEADS	correlationIdIn comp	
14	getLOsByMetadataEADS	correlationIdIn metadata values	
15	getLOsForCompetenciesEADSUsingStrategies	correlationIdIn strategy	
16	getLOsForCompetenciesEADS	correlationIdIn comp_level	
17	getLOsForCompetenciesUHP	correlationIdIn comp_level	
18	evaluateCorrelationId	correlationIdIn	
19	getTentativeComposition	correlationIdIn compositionRequest	
20	saveCompositionForTutor	correlationIdIn saveCompositionRequest	
21	packagesSavedForTutor	correlationIdIn	
22	deletePackage	correlationIdIn LOId	

Figure 10. Methods of the second prototype of the Negotiation Layer

The typical methods involved in a query are the ones related to getting LO. The Negotiation Layer WS methods have been customized for the case studies following a façade pattern, where a component exposes the required functionality in a friendly way. In this sense, the GUI may for instance retrieve LOs for EADS using competencies as input parameters (getLOsForCompetenciesEADS) or using strategies such as “hole in the profile”, “weakest field”, etc (getLOsForCompetenciesEADSUsingStrategies).

5.3.3 Query Resolvers

A Query Resolver implements specific organizational learning rules. The Query resolvers are thus use case-driven components, implementing functionalities that range from the preparation of the query by taking into account the user context, to the reception, processing and ranking of the results of the invocation.

For the second prototype we have provided with one Query Resolver for UHP and 5 for EADS that implement the business rules specified in the use cases.

- EADS Query Resolvers
 - Basic: Selection of competencies (default). In this case all competencies are selectable (even the ones that does not belong to the current profession) and the required levels can be changed respect to the current level. This Query Resolver was implemented for the first prototype.
 - All negative gaps: The query will get LOs to fulfil only the negative gaps
 - Weak field: The query will get LOs to fulfil only the weak gap
 - Hole in the profile : The query will get LOs to fulfil only the holes into the profile
 - Job specific: The query will get LOs specific for the user position
- UHP Query Resolver: Selection by the competencies selected by the user, their sub-competencies, and related topics.

More information about the Query Resolvers can be found in deliverables D6.3.2. and D7.3.2.

The Negotiation Layer implementation allows the inclusion of new Query Resolvers in the architecture by developing a simple Java class that implements a common Query Resolver interface:

Main Query Resolver methods:

- setUpQuery: This method loads the user query parameters plus the contextual preferences into the Query Resolver.
- parseXML: This method is in charge of parsing to java the list of LOs that returns the IRS-III in XML format.
- refineQueryRules: Applies the business rules over the list of retrieved LOs returned by the IRS-III previously parsed.
- rankingRules: Once the rules have been applied, this method implements the rules of ranking of a given organization.

5.3.4 Composition of LOs

The Composition component is in charge of creating new composed Learning Objects in a unique package. This package will be stored as a new LO in the LUISA LOMR and sent back to the Negotiation or the Query Resolver components. The management of these LOs (creation or deletion of LOs) will be handled by the Composition component.

Composition of LOs is not an easy task. Learning object composition is the process of combining several LOs into a higher level instructional unit that fulfils specific learning goals under concrete contextual circumstances. Composition requires selection as a constituent process. Nonetheless, composition imposes additional constraints coming from the fact that the LOs retrieved from different sub-goals can not be joined together. For instance, due to incompatible style designs or even to inconsistency in the pedagogical approach or level of description. There are a lot of different strategies, algorithms and possibilities for learning object composition. The idea of the Composition component is then to offer a unique interface for composition of LOs, but giving room to different concrete implementations of the algorithm for this automation. As in the case of the Query Resolver, we expect multiple “instances” of the Composition component that can be manually or automatically selected.

Thus, a set of different composers can be implemented following different strategies. In the first version of the prototype we delivered a simple automatic Composition component, valid for the two case studies. The composition proved to be not very useful. For the second prototype a more sophisticated composition of pedagogical packages has been developed for the UHP case study, reusing the previous composition service in a smarter way. More details can be found in deliverable D7.3.2.

6 ADDITIONAL OUTCOMES

Apart from the outcomes proposed here and in D2.3.1 [8], some additional results had been obtained as joint work between WP2 of LUISA and related EU projects, such as SUPER² and LHDL³.

While working on approaches to R3 (Negotiation), R4 (Context-Awareness) and R5 (Fuzzy Matchmaking), we identified the issue, that end-users usually are not able to describe their current need in a formal way complying with WSMO. Hence, the question was, how to support a rather user-driven method of eliciting WSMO goals from non-experts. In order to address this issue, an approach based on Wikis is proposed in [24], which enables the incremental refinement of goals by users with the aid of SWS engineers, leading from an informal goal description to a formally specified WSMO goal which is achievable in terms of SWS invocations. Hence, instead of a priori specifications of WSMO goals by experts, this approach facilitates the collaborative, user-driven and incremental refinement of WSMO goals by applying principles from the agile software development movement and taking advantage of Wiki technology.

With respect to R4, we also consider to deal with low-level Web service invocation contexts, which are important parameters when discovering the most appropriate service for a given process context. As a consequence of increasingly automated process, as proposed here, it is significantly difficult to reach a deep understanding on how things are actually performed, whether they were executed correctly or whether they can be improved. In order to support the analysis of processes both at runtime and post-execution, we have defined a set of ontologies that capture details encompassing low-level monitoring information (such as messages exchanges) and high-level details about processes like the agents involved, or even analysis results. Even though these ontologies initially had been applied to the business process domain [31] they will become an implicit part of the ontology stack proposed in Section 4.

With respect to R2 ("Domain-specificity") and R6 ("Interoperability"), further work was related to applying the conceptual model and ontology stack (Sections 3 and 4) of LUISA to the business process domain. There, another ontology, the Business Process Modelling Ontology (BPMO), was created, which allows business analysts to model business process workflows, similarly to the LPMO for the eLearning domain. Analogous to LPMO, BPMO uses SWS to represent Tasks/Activities in processes, in particular WSMO Goals. Moreover, transformation facilities had been provided to transform BPMO process instances in business process metadata standards at the execution level to enable interoperability. A preliminary prototype⁴ has been implemented using IRS-III.

² <http://www.ip-super.org/>

³ <http://www.livinghuman.org/>

⁴ Tutorial on Semantic Business Process Management at the 6th International Semantic Web Conference ISWC 2007 November 11-15 Busan, South Korea.

Continuing from this work on aligning the learning-related process ontologies of LUISA with BPMO, a process algebraic behavioural semantics is given by the Behavioural Reasoning Ontology (BRO). BRO is based on an ontologisation of the process algebra CaSE [28]. As well as an extra-ontological operational semantics via CaSE, ontology-based reasoning over the algebraic semantics is being developed as an axiomatisation of BRO. The translation between BPMO and BRO is furthermore carried out automatically, as WSMO data mediation, using ontological rules.

Since with respect to SWS discovery, the selected Web service (WS) is not known *a priori* by the user, the notion of trust is an important aspect of WS selection and highly context-dependent (R4), i.e. the same user may have different trust preferences in different contexts. In order to address this aspect, we propose a general approach for managing trust in SWS. We have developed an ontology – Web Services Trust-management Ontology (WSTO) [15] - which is able to represent trust specifications within SWS based on WSMO. In WSTO, we characterize trust-based Web service selection as a classification problem – i.e., given a set of user and Web service trust requirements and guarantees, our goal is to identify the class of Web services that match the trust statements of involved interaction participants, according to an established classification criterion.

7 CONCLUSIONS

To address the limitations of current E-Learning technologies, LUISA follows the vision of a Semantic Web Service (SWS) based infrastructure. We base such an infrastructure on WSMO, as reference ontology for semantically describing services, and IRS-III as reference SWS-environment. To tailor such an infrastructure to the specific needs of the E-Learning domain, several requirements had been identified (D.2.2), for instance the integration of heterogeneous learning resources, the domain-specific derivation of WSMO and the need for dynamic interactions with the SWS layer.

This document – together with D2.3.1 [8] - proposes our implementation approach followed to address these requirements which is fundamentally based on a conceptual framework foreseeing the semantic abstraction from learning resources – data and services – and the incorporation of semantic representations into comprehensive domain contexts. This is realized by providing additional software components and semantic representations which support our vision and conceptual approach. For each of the identified requirements, a set of implementation activities has been described which addresses the requirement by supporting the needs of the LUISA use cases (WP6, WP7). To enable integration of heterogeneous resources, our implementation approach based on IRS-III and WSMO was elaborated. Domain-specificity is supported through the gradual derivation of WSMO for the E-Learning domain based on a stack of ontologies. These include the Learning Process Modelling Ontology (LPMO), the Situation-driven Process Ontology (SDPO) and set of use case related ontologies and SWS description. The ontology stack not only enables the domain-specific derivation of WSMO but also the representation comprehensive domain contexts, since the LPMO provides a formal semantic conceptualization of the E-Learning domain. Interactive negotiations during SWS orchestration is enabled through the implementation of a specific architectural component, the Negotiation Layer.

Fuzzy matchmaking between real-world E-Learning context situations and predefined semantic context representations based on LPMO is supported through the so-called Conceptual Situation Spaces – presented in D2.3.1 [8] - which enable similarity-based matchmaking. Finally, interoperability with established metadata standards is supported through aligning the existing ontology framework to semantic representations of established metadata standards, such as IMS LD and ADL SCORM. Details regarding metadata interoperability are presented in [8].

Since some of the requirements (R1 – R3, Section 2) are derived directly from the use cases (WP6, WP7), these are addressed through components which are already part of the overall LUISA prototype application. In contrast, other requirements for adoption of WSMO (R4 - R6) are derived from the E-Learning domain in general and are currently addressed through semantic facilities and software components of a separate prototype application which has been an additional result of WP2.

Whereas some of the provided facilities are developed for specific use cases, the entire LUISA application framework is perceived as a general-purpose SWS

architecture for the E-Learning domain which could be deployed across further application settings and use case scenarios.

REFERENCES

- [1] Advanced Distributed Learning (ADL) SCORM 2004 Specification (<http://www.adlnet.org>).
- [2] Amorim, R. R., Lama, M., Sánchez, E., Riera, A., & Vila, X. A. (2006). A Learning Design Ontology based on the IMS Specification. *Journal of Educational Technology & Society*, 9 (1), 38-57.
- [3] Cabral, L., Domingue, J., Galizia, S., Gugliotta, A., Norton, B., Tanasescu, V., Pedrinaci, C.: IRS-III: A Broker for Semantic Web Services based Applications. In proceedings of the 5th International Semantic Web Conference (ISWC 2006), Athens, USA (2006).
- [4] Collis, B. and Strijker, A. (2004). Technology and Human Issues in Reusing Learning Objects. *Journal of Interactive Media in Education*, 2004 (4). Special Issue on the Educational Semantic Web [[www-jime.open.ac.uk/2004/4](http://www.jime.open.ac.uk/2004/4)].
- [5] Cregan, A. (2007), Symbol Grounding for the Semantic Web. 4th European Semantic Web Conference 2007, Innsbruck, Austria.
- [6] De Bruijn, J., Lausen, H., Krummenacher, R., Polleres, A., Predoiu, L., Kifer, M., Fensel, D.: Web service Modeling Language (WSML). Final Draft (2005). (<http://www.wsmo.org/TR/d16/d16.1/v0.21/20051005/>).
- [7] Dietze, S., Gugliotta, A., Domingue, J.: A Semantic Web Service oriented Framework for adaptive Learning Environments, In Proceedings of the 4th European Semantic Web Conference, Innsbruck, Austria, (2007).
- [8] Dietze, S., Gugliotta, A., Domingue, J., Pariente Lobo, T. (2008). Semantic Web Service Infrastructure Implementation – Publication, Discovery, Execution, LUISA Deliverable D2.3.1, LUISA project (2008).
- [9] Domingue, J., Motta, E., and Corcho Garcia, O. (1999). Knowledge Modelling in WebOnto and OCML: A User Guide, available from: http://kmi.open.ac.uk/projects/webonto/user_guide.2.4.pdf.
- [10] Duval, E. (2002). 1484.12.1 IEEE Standard for learning Object Metadata, IEEE Learning Technology Standards Committee, <http://ltsc.ieee.org/wg12/>.
- [11] Felder, R. M., Silverman, L. K. (1988). Learning and Teaching Styles in Engineering Education, *Engineering Education*, 78, 1988, pp. 674–681. Preceded by a preface in 2002: <http://www.ncsu.edu/felderpublic/Papers/LS-1988.pdf>.
- [12] Felder, R. M. and Soloman, L. K. (1997) Index of Learning Styles Questionnaire. Online version <http://www.engr.ncsu.edu/learningstyles/ilsweb.html> (retrieved October, 2007).
- [13] Felder, R.M. and Spurlin, J., (2005). Applications, Reliability and Validity of the Index of Learning Styles. *International Journal on Engineering Education*, Vol. 21, No. 1, pp. 103-112.
- [14] Fensel, D., Lausen, H., Polleres, A., de Bruijn, J., Stollberg, M., Roman, D., Domingue, J. : Enabling Semantic Web Services – The Web service Modelling Ontology, Springer 2006.

- [15] Galizia, S., (2006) WSTO: A Classification-Based Ontology for Managing Trust in Semantic Web Services. 3th European Semantic Web Conference, Budva (Montenegro).
- [16] Gangemi, A., and Mika, P. Understanding the Semantic Web through Descriptions and Situations. In Meersman, R.; Tari, Z.; and et al., D. S., eds., Proceedings of the On The Move Federated Conferences (OTM'03), LNCS. Springer Verlag.
- [17] Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening Ontologies with DOLCE. In: A. Gómez-Pérez , V. Richard Benjamins (Eds.) Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web: 13th International Conference, EKAW 2002, Siguenza, Spain, October 1-4, 2002.
- [18] Gärdenfors, P. (2000), Conceptual Spaces - The Geometry of Thought. MIT Press, 2000.
- [19] Gärdenfors, P. (2004), How to make the semantic web more semantic. In A.C. Vieu and L. Varzi, editors, Formal Ontology in Information Systems, pages 19–36. IOS Press, 2004.
- [20] Graf, S., Viola, S.R., Kinshuk, and Leo, T. (2006). Representative Characteristics of Felder-Silverman Learning Styles: An Empirical Model. Proceedings of the IADIS International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2006), Barcelona, Spain, December 8-10.
- [21] IMS Learning Design Specification (<http://www.imsglobal.org>).
- [22] Johnson, C., Orwig, C.(1998), What is Learning Style?, <http://www.sil.org/LinguaLinks/Language-Learning/OtherResources/YorLrningStylAndLnggLrnng/WhatIsALearningStyle.htm>.
- [23] Knight, C., Gašević, D., & Richards, G. (2006). An Ontology-Based Framework for Bridging Learning Design and Learning Content. Journal of Educational Technology & Society, 9 (1), 23-37.
- [24] Lambert, D., Galizia, S., Domingue, J., (2007) Agile elicitation of semantic goals by wiki. Workshop: Human-friendly (Web) Service Description, Discovery and Matchmaking, The 8th International Conference on Web Information Systems Engineering, Nancy, France.
- [25] LUISA Consortium (2008) D6.3.2 EADS use case implementation and prototyping
- [26] LUISA Consortium (2008) D7.3.2 UHP use case implementation and prototyping
- [27] Motta, E. (1998). An Overview of the OCML Modelling Language, the 8th Workshop on Methods and Languages, 1998.
- [28] Norton, B., L A Compositional Semantic Theory for Synchronous Component-Based Design uettgen, G., Mender, M. (2003), in 14th Intl. Conference on Concurrency Theory (CONCUR'03), Springer-Verlag, 2003.
- [29] Nosofsky, R. (1992), Similarity, scaling and cognitive process models, Annual Review of Psychology 43, pp. 25- 53, (1992).

- [30] OWL-S 1.0 Release. <http://www.daml.org/services/owl-s/1.0/>.
- [31] Pedrinaci, C., Domingue, J., Karla, A., (2008) A Core Ontology for Business Process Analysis. 5th European Semantic Web Conference 2008, Tenerife, Spain.
- [32] Raubal, M. (2004). Formalizing Conceptual Spaces. in: A. Varzi and L. Vieu (Eds.), Formal Ontology in Information Systems, Proceedings of the Third International Conference (FOIS 2004).Frontiers in Artificial Intelligence and Applications 114, pp. 153-164, IOS Press, Amsterdam, NL.
- [33] Reload Project (<http://www.reload.ac.uk/>).
- [34] World Wide Web Consortium, W3C: Simple Object Access Protocol, SOAP, Version 1.2 Part 0: Primer, (2003). (<http://www.w3.org/TR/soap12-part0/>).
- [35] Viola, S.R., Graf, S., Kinshuk, and Leo T. (2007). Investigating Relationships within the Index of Learning Styles: A Data-Driven Approach. International Journal of Interactive Technology and Smart Education, Vol. 4, No. 1, pp. 7-18.
- [36] World Wide Web Consortium, W3C: Universal Description, Discovery and Integration: UDDI Spec Technical Committee Specification v. 3.0, (2003). <http://uddi.org/pubs/uddi-v3.0.1-20031014.htm>.
- [37] World Wide Web Consortium, W3C: WSDL: Web services Description Language (WSDL) 1.1, (2001). (<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>).
- [38] WSMO Working Group, D2v1.0: Web service Modeling Ontology (WSMO). WSMO Working Draft, (2004). (<http://www.wsmo.org/2004/d2/v1.0/>).
- [39] WSMO Working Group: WSMO Studio SWS Editor. (<http://www.wsmostudio.org/>).
- [40] WSMO Working Group, D2v1.0: Web service Modeling Ontology (WSMO). WSMO Working Draft, (2004). (<http://www.wsmo.org/2004/d2/v1.0/>).

8 APPENDIX

8.1 SWS Descriptions aimed at retrieving (EADS) LOM given a specific LOM field

```
;; Meta goal including orchestration

(DEF-CLASS GET-EADS-LO-BY-METADATA-GOAL-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS Get-EADS-LO-BY-METADATA-GOAL
(GOAL)
?GOAL
((HAS-INPUT-ROLE :VALUE has-collection
:VALUE has-search
:VALUE has-attributes
:VALUE has-language)
(HAS-INPUT-SOAP-BINDING
:VALUE (has-collection "string")
:VALUE (has-search "string")
:VALUE (has-language "string")
:VALUE (has-attributes "string"))
(HAS-OUTPUT-ROLE :VALUE has-lo-metadata)
(HAS-OUTPUT-SOAP-BINDING :VALUE (has-lo-metadata "xml"))
(has-collection :type http-string)
(has-search :type string)
(has-language :type http-string)
(has-attributes :type string)
(has-lo-metadata :type string);lo-metadata-list)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
Get-EADS-LO-BY-METADATA-GOAL-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS Get-EADS-LO-BY-METADATA-MED-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS Get-EADS-LO-BY-METADATA-MED
(WG-MEDIATOR)
?MEDIATOR
((HAS-SOURCE-COMPONENT :VALUE Get-EADS-LO-BY-METADATA-GOAL)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
Get-EADS-LO-BY-METADATA-MED-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS Get-EADS-LO-BY-METADATA-WS-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS Get-EADS-LO-BY-METADATA-WS
(WEB-SERVICE)
?WEB-SERVICE
((HAS-CAPABILITY :VALUE Get-EADS-LO-BY-METADATA-WS-CAPABILITY)
(HAS-INTERFACE :VALUE Get-EADS-LO-BY-METADATA-INTERFACE)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
Get-EADS-LO-BY-METADATA-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS Get-EADS-LO-BY-METADATA-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS Get-EADS-LO-BY-METADATA-WS-CAPABILITY
(CAPABILITY)
?CAPABILITY
((USED-MEDIATOR :VALUE Get-EADS-LO-BY-METADATA-MED)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
```

```

        Get-EADS-LO-BY-METADATA-WS-CAPABILITY-NON-FUNCTIONAL-
        PROPERTIES)))
(DEF-CLASS Get-EADS-LO-BY-METADATA-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)
(DEF-CLASS Get-EADS-LO-BY-METADATA-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING :VALUE ((NORMAL DUMMY-FUNCTION))))))
(DEF-CLASS Get-EADS-LO-BY-METADATA-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  ((HAS-BODY :VALUE ((ORCH-SEQ MULTI-GET-EADS-LO-REF-GOAL GET-EADS-LO-
METADATA-GOAL))))))
(DEF-CLASS Get-EADS-LO-BY-METADATA-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
:VALUE
  Get-EADS-LO-BY-METADATA-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN)))
(DEF-CLASS Get-EADS-LO-BY-METADATA-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
:VALUE
  Get-EADS-LO-BY-METADATA-INTERFACE-CHOREOGRAPHY)
(HAS-ORCHESTRATION
:VALUE
  Get-EADS-LO-BY-METADATA-INTERFACE-ORCHESTRATION)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
  Get-EADS-LO-BY-METADATA-INTERFACE-NON-FUNCTIONAL-PROPERTIES)))
;; first goal of orchestration
(DEF-CLASS MULTI-GET-EADS-LO-REF-GOAL-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)
(DEF-CLASS MULTI-GET-EADS-LO-REF-GOAL
  (GOAL)
  ?GOAL
  ((HAS-INPUT-ROLE :VALUE has-collection
:VALUE has-search)
(HAS-INPUT-SOAP-BINDING :VALUE (has-collection "string")
:VALUE (has-search "string"))
(HAS-OUTPUT-ROLE :VALUE has-lo-refs)
(HAS-OUTPUT-SOAP-BINDING :VALUE (has-lo-refs "xml"))
(has-search :type string)
(has-collection :type http-string)
(has-lo-refs :TYPE multi-eads-lo-ref-string)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
  MULTI-GET-EADS-LO-REF-GOAL-NON-FUNCTIONAL-PROPERTIES)))
(DEF-CLASS MULTI-GET-EADS-LO-REF-MEDIATOR-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)
(DEF-CLASS MULTI-GET-EADS-LO-REF-MEDIATOR
  (WG-MEDIATOR)
  ?MEDIATOR
  ((HAS-SOURCE-COMPONENT :VALUE MULTI-GET-EADS-LO-REF-GOAL)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
  MULTI-GET-EADS-LO-REF-MED-NON-FUNCTIONAL-PROPERTIES)))
;; web service by keyword
(DEF-CLASS KEYWORD-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

```

```
(DEF-CLASS KEYWORD-GET-EADS-LO-REF-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE KEYWORD-GET-EADS-LO-REF-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE KEYWORD-GET-EADS-LO-REF-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    KEYWORD-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-WS-CAPABILITY
  (CAPABILITY)
  ?CAPABILITY
  ((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
   (HAS-ASSUMPTION
    :VALUE
    (KAPPA
     (?WEB-SERVICE)
     (= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
        "keyword"))))
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE KEYWORD-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING
   :VALUE
   ((GROUNDED-TO-HTTP
    (NORMAL
     (NIL (collection) "eadsLOsearchByKeyword"))))))))

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  NIL)

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
   :VALUE
   KEYWORD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
   PATTERN)))

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
   :VALUE KEYWORD-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY)
   (HAS-ORCHESTRATION
   :VALUE KEYWORD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION)
   (HAS-NON-FUNCTIONAL-PROPERTIES
   :VALUE KEYWORD-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-
   PROPERTIES)))

(DEF-CLASS KEYWORD-GET-EADS-LO-REF-WS-PUBLISHER-INFORMATION
  (PUBLISHER-INFORMATION)
  ((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
   :VALUE
   KEYWORD-GET-EADS-LO-REF-INTERFACE)
   (HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
   (HAS-WEB-SERVICE-PORT :VALUE 8080)
   (HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSEads.jws")))

;; web service by code
(DEF-CLASS CODE-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)
```

```
(DEF-CLASS CODE-GET-EADS-LO-REF-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE CODE-GET-EADS-LO-REF-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE CODE-GET-EADS-LO-REF-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    CODE-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS CODE-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS CODE-GET-EADS-LO-REF-WS-CAPABILITY
  (CAPABILITY)
  ?CAPABILITY
  ((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
   (HAS-ASSUMPTION
    :VALUE
    (KAPPA
     (?WEB-SERVICE)
     (= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
        "code"))))
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE CODE-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS CODE-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS CODE-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING
   :VALUE
   ((GROUNDED-TO-HTTP
    (NORMAL
     (NIL (collection) "eadsLOsearchByCode"))))))))

(DEF-CLASS CODE-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  NIL)

(DEF-CLASS CODE-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
   :VALUE
   CODE-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
   PATTERN)))

(DEF-CLASS CODE-GET-EADS-LO-REF-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
   :VALUE CODE-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY)
   (HAS-ORCHESTRATION
   :VALUE CODE-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION)
   (HAS-NON-FUNCTIONAL-PROPERTIES
   :VALUE CODE-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-
   PROPERTIES)))

(DEF-CLASS CODE-GET-EADS-LO-REF-WS-PUBLISHER-INFORMATION
  (PUBLISHER-INFORMATION)
  ((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
   :VALUE
   CODE-GET-EADS-LO-REF-INTERFACE)
   (HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
   (HAS-WEB-SERVICE-PORT :VALUE 8080)
   (HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSEads.jws")))

;; web service by function

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES
```

```

(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE FUNCTION-GET-EADS-LO-REF-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE FUNCTION-GET-EADS-LO-REF-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    FUNCTION-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-WS-CAPABILITY
  (CAPABILITY)
  ?CAPABILITY
  ((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
   (HAS-ASSUMPTION
    :VALUE
    (KAPPA
     (?WEB-SERVICE
      (= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
         "function"))))
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE FUNCTION-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING
   :VALUE
   ((GROUNDED-TO-HTTP
    (NORMAL
     (NIL (collection) "eadsLOsearchByFunction"))))))))

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  NIL)

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
   :VALUE
   FUNCTION-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
   PATTERN)))

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
   :VALUE FUNCTION-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY)
   (HAS-ORCHESTRATION
   :VALUE FUNCTION-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION)
   (HAS-NON-FUNCTIONAL-PROPERTIES
   :VALUE FUNCTION-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-
   PROPERTIES)))

(DEF-CLASS FUNCTION-GET-EADS-LO-REF-WS-PUBLISHER-INFORMATION
  (PUBLISHER-INFORMATION)
  ((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
   :VALUE
   FUNCTION-GET-EADS-LO-REF-INTERFACE)
   (HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
   (HAS-WEB-SERVICE-PORT :VALUE 8080)
   (HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSEads.jws")))

```

```
;; web service by field
```

```
(DEF-CLASS FIELD-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS FIELD-GET-EADS-LO-REF-WS
(WEB-SERVICE)
?WEB-SERVICE
((HAS-CAPABILITY :VALUE FIELD-GET-EADS-LO-REF-WS-CAPABILITY)
(HAS-INTERFACE :VALUE FIELD-GET-EADS-LO-REF-INTERFACE)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
FIELD-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS FIELD-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS FIELD-GET-EADS-LO-REF-WS-CAPABILITY
(CAPABILITY)
?CAPABILITY
((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
(HAS-ASSUMPTION
:VALUE
(KAPPA
(?WEB-SERVICE)
(= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
"field"))))
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE FIELD-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-
PROPERTIES)))

(DEF-CLASS FIELD-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS FIELD-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY
(CHOREOGRAPHY)
((HAS-GROUNDING
:VALUE
((GROUNDED-TO-HTTP
(NORMAL
(NIL (collection) "eadsLOsearchByField"))))))))

(DEF-CLASS FIELD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
NIL)

(DEF-CLASS FIELD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION
(ORCHESTRATION)
((HAS-PROBLEM-SOLVING-PATTERN
:VALUE
FIELD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN)))

(DEF-CLASS FIELD-GET-EADS-LO-REF-INTERFACE
(INTERFACE)
?INTERFACE
((HAS-CHOREOGRAPHY
:VALUE FIELD-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY)
(HAS-ORCHESTRATION
:VALUE FIELD-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE FIELD-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-
PROPERTIES)))

(DEF-CLASS FIELD-GET-EADS-LO-REF-WS-PUBLISHER-INFORMATION
(PUBLISHER-INFORMATION)
((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
:VALUE
FIELD-GET-EADS-LO-REF-INTERFACE)
(HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
(HAS-WEB-SERVICE-PORT :VALUE 8080)
(HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSEads.jws")))
```

```

;; web service by domain

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE DOMAIN-GET-EADS-LO-REF-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE DOMAIN-GET-EADS-LO-REF-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    DOMAIN-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-WS-CAPABILITY
  (CAPABILITY)
  ?CAPABILITY
  ((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
   (HAS-ASSUMPTION
    :VALUE
    (KAPPA
     (?WEB-SERVICE)
     (= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
        "domain"))))
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE DOMAIN-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING
    :VALUE
    ((GROUNDED-TO-HTTP
     (NORMAL
      (NIL (collection) "eadsLosearchByDomain"))))))))

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  NIL)

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
    :VALUE
    DOMAIN-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
    PATTERN)))

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
    :VALUE DOMAIN-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY)
   (HAS-ORCHESTRATION
    :VALUE DOMAIN-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE DOMAIN-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS DOMAIN-GET-EADS-LO-REF-WS-PUBLISHER-INFORMATION
  (PUBLISHER-INFORMATION)
  ((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
    :VALUE
    DOMAIN-GET-EADS-LO-REF-INTERFACE)
   (HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
   (HAS-WEB-SERVICE-PORT :VALUE 8080)
   (HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSEads.jws")))

```

```
;; web service by speciality

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE SPECIALITY-GET-EADS-LO-REF-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE SPECIALITY-GET-EADS-LO-REF-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    SPECIALITY-GET-EADS-LO-REF-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-WS-CAPABILITY
  (CAPABILITY)
  ?CAPABILITY
  ((USED-MEDIATOR :VALUE MULTI-GET-EADS-LO-REF-MEDIATOR)
   (HAS-ASSUMPTION
    :VALUE
    (KAPPA
     (?WEB-SERVICE
      (= (WSMO-ROLE-VALUE ?WEB-SERVICE 'has-search)
         "speciality"))))
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE SPECIALITY-GET-EADS-LO-REF-WS-CAPABILITY-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING
    :VALUE
    ((GROUNDED-TO-HTTP
     (NORMAL
      (NIL (collection) "eadsLOsearchBySpeciality"))))))))

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  NIL)

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
    :VALUE
    SPECIALITY-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION-PROBLEM-
    SOLVING-PATTERN)))

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
    :VALUE SPECIALITY-GET-EADS-LO-REF-INTERFACE-CHOREOGRAPHY)
   (HAS-ORCHESTRATION
    :VALUE SPECIALITY-GET-EADS-LO-REF-INTERFACE-ORCHESTRATION)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE SPECIALITY-GET-EADS-LO-REF-INTERFACE-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS SPECIALITY-GET-EADS-LO-REF-WS-PUBLISHER-INFORMATION
  (PUBLISHER-INFORMATION)
  ((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
    :VALUE
    SPECIALITY-GET-EADS-LO-REF-INTERFACE)
   (HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
   (HAS-WEB-SERVICE-PORT :VALUE 8080))
```

```
(HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSEads.jws"))
```

8.2 SWS Descriptions aimed at retrieving (UHP) LOM by Topic

```
;;; meta-goal including orchestration

(DEF-CLASS GET-UHP-LO-BY-TOPIC-GOAL-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-BY-TOPIC-GOAL
  (GOAL)
  ?GOAL
  ((HAS-INPUT-ROLE :VALUE has-topics
                   :VALUE has-attributes
                   :VALUE has-language)
   (HAS-INPUT-SOAP-BINDING
    :VALUE (has-topics "string")
    :VALUE (has-language "string")
    :VALUE (has-attributes "string"))
   (HAS-OUTPUT-ROLE :VALUE has-lo-metadata)
   (HAS-OUTPUT-SOAP-BINDING :VALUE (has-lo-metadata "xml"))
   (has-topics :type http-string)
   (has-language :type http-string)
   (has-attributes :type string)
   (has-lo-metadata :type string);lo-metadata-list)
  (HAS-NON-FUNCTIONAL-PROPERTIES
   :VALUE
   GET-UHP-LO-BY-TOPIC-GOAL-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-MED-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-BY-TOPIC-MED
  (WG-MEDIATOR)
  ?MEDIATOR
  ((HAS-SOURCE-COMPONENT :VALUE GET-UHP-LO-BY-TOPIC-GOAL)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    GET-UHP-LO-BY-TOPIC-MED-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-WS-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-BY-TOPIC-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE GET-UHP-LO-BY-TOPIC-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE GET-UHP-LO-BY-TOPIC-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    GET-UHP-LO-BY-TOPIC-WS-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-BY-TOPIC-WS-CAPABILITY
  (CAPABILITY)
  ?CAPABILITY
  ((USED-MEDIATOR :VALUE GET-UHP-LO-BY-TOPIC-MED)
   (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE GET-UHP-LO-BY-TOPIC-WS-CAPABILITY-NON-FUNCTIONAL-
    PROPERTIES)))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-INTERFACE-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)
```

```
(DEF-CLASS GET-UHP-LO-BY-TOPIC-INTERFACE-CHOREOGRAPHY
  (CHOREOGRAPHY)
  ((HAS-GROUNDING :VALUE ((NORMAL DUMMY-FUNCTION))))))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-PATTERN
  (PROBLEM-SOLVING-PATTERN)
  ((HAS-BODY :VALUE ((ORCH-SEQ GET-UHP-LO-REF-BY-TOPIC-GOAL
    GET-UHP-LO-METADATA-GOAL))))))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-INTERFACE-ORCHESTRATION
  (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN
    :VALUE
    GET-UHP-LO-BY-TOPIC-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
    PATTERN)))

(DEF-CLASS GET-UHP-LO-BY-TOPIC-INTERFACE
  (INTERFACE)
  ?INTERFACE
  ((HAS-CHOREOGRAPHY
    :VALUE
    GET-UHP-LO-BY-TOPIC-INTERFACE-CHOREOGRAPHY)
  (HAS-ORCHESTRATION
    :VALUE
    GET-UHP-LO-BY-TOPIC-INTERFACE-ORCHESTRATION)
  (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    GET-UHP-LO-BY-TOPIC-INTERFACE-NON-FUNCTIONAL-PROPERTIES)))

;; first goal of the orchestration

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-GOAL-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-GOAL
  (GOAL)
  ?GOAL
  ((HAS-INPUT-ROLE :VALUE has-topics)
  (HAS-INPUT-SOAP-BINDING :VALUE (has-topics "string"))
  (HAS-OUTPUT-ROLE :VALUE has-lo-refs)
  (HAS-OUTPUT-SOAP-BINDING :VALUE (has-lo-refs "xml"))
  (has-lo-refs :TYPE uhp-lo-ref-string-by-topic)
  (has-topics :TYPE http-string)
  (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    GET-UHP-LO-REF-BY-TOPIC-GOAL-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-MED-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-MED
  (WG-MEDIATOR)
  ?MEDIATOR
  ((HAS-SOURCE-COMPONENT :VALUE GET-UHP-LO-REF-BY-TOPIC-GOAL)
  (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    GET-UHP-LO-REF-BY-TOPIC-MED-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-WS-NON-FUNCTIONAL-PROPERTIES
  (NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-WS
  (WEB-SERVICE)
  ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE GET-UHP-LO-REF-BY-TOPIC-WS-CAPABILITY)
  (HAS-INTERFACE :VALUE GET-UHP-LO-REF-BY-TOPIC-INTERFACE)
  (USED-MEDIATOR :VALUE GET-UHP-LO-REF-BY-TOPIC-MED)
  (HAS-NON-FUNCTIONAL-PROPERTIES
    :VALUE
    GET-UHP-LO-REF-BY-TOPIC-WS-NON-FUNCTIONAL-PROPERTIES)))
```

```
(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-WS-CAPABILITY
(CAPABILITY)
?CAPABILITY
((USED-MEDIATOR :VALUE GET-UHP-LO-REF-BY-TOPIC-MED)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
GET-UHP-LO-REF-BY-TOPIC-WS-CAPABILITY-NON-FUNCTIONAL-
PROPERTIES)))

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-INTERFACE-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-INTERFACE-CHOREOGRAPHY
(CHOREOGRAPHY)
((HAS-GROUNDING
:VALUE
((GROUNDED-TO-HTTP
(NORMAL
(NIL ("topic") "uhpLOsearchByTopic"))))))))

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
NIL)

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-INTERFACE-ORCHESTRATION
(ORCHESTRATION)
((HAS-PROBLEM-SOLVING-PATTERN
:VALUE
GET-UHP-LO-REF-BY-TOPIC-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN)))

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-INTERFACE
(INTERFACE)
?INTERFACE
((HAS-CHOREOGRAPHY
:VALUE
GET-UHP-LO-REF-BY-TOPIC-INTERFACE-CHOREOGRAPHY)
(HAS-ORCHESTRATION
:VALUE
GET-UHP-LO-REF-BY-TOPIC-INTERFACE-ORCHESTRATION)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
GET-UHP-LO-REF-BY-TOPIC-INTERFACE-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-UHP-LO-REF-BY-TOPIC-WS-PUBLISHER-INFORMATION
(PUBLISHER-INFORMATION)
((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
:VALUE
GET-UHP-LO-REF-BY-TOPIC-INTERFACE)
(HAS-WEB-SERVICE-HOST :VALUE "secse.atosorigin.es")
(HAS-WEB-SERVICE-PORT :VALUE 8080)
(HAS-WEB-SERVICE-LOCATION :VALUE "/axis/WSUhp.jws")))

```

8.3 SWS Descriptions aimed at retrieving both UHP and EADS LOM by Competency

```
;; meta goal including orchestration

(DEF-CLASS META-GET-LO-BY-COMP-GOAL-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS META-GET-LO-BY-COMP-GOAL
(GOAL)
?GOAL
((HAS-INPUT-ROLE :VALUE has-meta-competencies
:VALUE has-values

```

```

:VALUE has-duration
:VALUE has-cost
:VALUE has-discipline
:VALUE has-language
:VALUE has-operating-system
:VALUE has-applications
:VALUE has-attributes)
(HAS-INPUT-SOAP-BINDING
:VALUE (has-meta-competencies "string")
:VALUE (has-values "string")
:VALUE (has-duration "string")
:VALUE (has-cost "string")
:VALUE (has-discipline "string")
:VALUE (has-language "string")
:VALUE (has-operating-system "string")
:VALUE (has-applications "string")
:VALUE (has-attributes "string"))
(HAS-OUTPUT-ROLE :VALUE has-lo-metadata)
(HAS-OUTPUT-SOAP-BINDING :VALUE (has-lo-metadata "xml"))
(has-meta-competencies :type http-string)
(has-values :type string)
(has-duration :type integer)
(has-cost :type string)
(has-discipline :type http-string)
(has-language :type http-string)
(has-operating-system :type http-string)
(has-applications :type http-string)
(has-attributes :type string)
(has-lo-metadata :type meta-lo-list)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
META-GET-LO-BY-COMP-GOAL-NON-FUNCTIONAL-PROPERTIES)) )

(DEF-CLASS META-GET-LO-BY-COMP-MED-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS META-GET-LO-BY-COMP-MED
(WG-MEDIATOR)
?MEDIATOR
((HAS-SOURCE-COMPONENT :VALUE META-GET-LO-BY-COMP-GOAL)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
META-GET-LO-BY-COMP-MED-NON-FUNCTIONAL-PROPERTIES)) )

(DEF-CLASS META-GET-LO-BY-COMP-WS-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS META-GET-LO-BY-COMP-WS
(WEB-SERVICE)
?WEB-SERVICE
((HAS-CAPABILITY :VALUE META-GET-LO-BY-COMP-WS-CAPABILITY)
(HAS-INTERFACE :VALUE META-GET-LO-BY-COMP-INTERFACE)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
META-GET-LO-BY-COMP-WS-NON-FUNCTIONAL-PROPERTIES)) )

(DEF-CLASS META-GET-LO-BY-COMP-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS META-GET-LO-BY-COMP-WS-CAPABILITY
(CAPABILITY)
?CAPABILITY
((USED-MEDIATOR :VALUE META-GET-LO-BY-COMP-MED)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE META-GET-LO-BY-COMP-WS-CAPABILITY-NON-FUNCTIONAL-
PROPERTIES)) )

(DEF-CLASS META-GET-LO-BY-COMP-INTERFACE-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS META-GET-LO-BY-COMP-INTERFACE-CHOREOGRAPHY

```

```

(CHOREOGRAPHY)
((HAS-GROUNDING :VALUE ((NORMAL DUMMY-FUNCTION))))

(DEF-CLASS META-GET-LO-BY-COMP-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-PATTERN
(PROBLEM-SOLVING-PATTERN)
((HAS-BODY :VALUE ((ORCH-SEQ GET-MAPPED-COMPETENCIES-GOAL)
(ORCH-SEQ (ACHIEVE-GOAL GET-UHP-LO-BY-COMP-GOAL
(first (ORCH-GET-GOAL-VALUE GET-MAPPED-COMPETENCIES-GOAL))
(orch-get-input-role 'ocml::has-values)
(orch-get-input-role 'ocml::has-duration)
(orch-get-input-role 'ocml::has-cost)
(orch-get-input-role 'ocml::has-discipline)
(orch-get-input-role 'ocml::has-language)
(orch-get-input-role 'ocml::has-operating-system)
(orch-get-input-role 'ocml::has-applications)
(orch-get-input-role 'ocml::has-attributes))
(ACHIEVE-GOAL GET-EADS-LO-BY-COMP-GOAL
(second (ORCH-GET-GOAL-VALUE GET-MAPPED-COMPETENCIES-GOAL))
(orch-get-input-role 'ocml::has-values)
(orch-get-input-role 'ocml::has-cost)
(orch-get-input-role 'ocml::has-language)
(orch-get-input-role 'ocml::has-duration)
(orch-get-input-role 'ocml::has-attributes)))
(list-of (ORCH-GET-GOAL-VALUE GET-UHP-LO-BY-COMP-GOAL)
(ORCH-GET-GOAL-VALUE GET-EADS-LO-BY-COMP-GOAL)
))))))

(DEF-CLASS META-GET-LO-BY-COMP-INTERFACE-ORCHESTRATION
(ORCHESTRATION)
((HAS-PROBLEM-SOLVING-PATTERN
:VALUE
META-GET-LO-BY-COMP-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN)))

(DEF-CLASS META-GET-LO-BY-COMP-INTERFACE
(INTERFACE)
?INTERFACE
((HAS-CHOREOGRAPHY
:VALUE
META-GET-LO-BY-COMP-INTERFACE-CHOREOGRAPHY)
(HAS-ORCHESTRATION
:VALUE
META-GET-LO-BY-COMP-INTERFACE-ORCHESTRATION)
(HAS-NON-FUNCTIONAL-PROPERTIES
:VALUE
META-GET-LO-BY-COMP-INTERFACE-NON-FUNCTIONAL-PROPERTIES)))

;; goal to map competencies (first goal of the orchestration)

(DEF-CLASS GET-MAPPED-COMPETENCIES-GOAL-NON-FUNCTIONAL-PROPERTIES (NON-
FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS GET-MAPPED-COMPETENCIES-GOAL (GOAL) ?GOAL
((HAS-INPUT-ROLE :VALUE HAS-META-COMPETENCIES)
(HAS-INPUT-SOAP-BINDING :VALUE (HAS-META-COMPETENCIES "string"))
(HAS-OUTPUT-ROLE :VALUE HAS-MATCHING-GENERALISATION)
(HAS-OUTPUT-SOAP-BINDING :VALUE (HAS-MATCHING-GENERALISATION "sexpr"))
(HAS-META-COMPETENCIES :TYPE HTTP-STRING)
(HAS-MATCHING-GENERALISATION :TYPE MATCHING-LIST)
(HAS-NON-FUNCTIONAL-PROPERTIES :VALUE GET-MAPPED-COMPETENCIES-GOAL-NON-
FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-MAPPED-COMPETENCIES-MED-NON-FUNCTIONAL-PROPERTIES (NON-
FUNCTIONAL-PROPERTIES)
NIL)

(DEF-CLASS GET-MAPPED-COMPETENCIES-MED (WG-MEDIATOR) ?MEDIATOR
((HAS-SOURCE-COMPONENT :VALUE GET-MAPPED-COMPETENCIES-GOAL)
(HAS-NON-FUNCTIONAL-PROPERTIES :VALUE GET-MAPPED-COMPETENCIES-MED-NON-
FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-MAPPED-COMPETENCIES-WS-NON-FUNCTIONAL-PROPERTIES (NON-
FUNCTIONAL-PROPERTIES)

```

```

NIL)

(DEF-CLASS GET-MAPPED-COMPETENCIES-WS (WEB-SERVICE) ?WEB-SERVICE
  ((HAS-CAPABILITY :VALUE GET-MAPPED-COMPETENCIES-WS-CAPABILITY)
   (HAS-INTERFACE :VALUE GET-MAPPED-COMPETENCIES-INTERFACE)
   (HAS-NON-FUNCTIONAL-PROPERTIES :VALUE GET-MAPPED-COMPETENCIES-WS-NON-
FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-MAPPED-COMPETENCIES-WS-CAPABILITY-NON-FUNCTIONAL-PROPERTIES
(NON-FUNCTIONAL-PROPERTIES)
  NIL)

(DEF-CLASS GET-MAPPED-COMPETENCIES-WS-CAPABILITY (CAPABILITY) ?CAPABILITY
  ((USED-MEDIATOR GET-MAPPED-COMPETENCIES-MED)
   (HAS-NON-FUNCTIONAL-PROPERTIES :VALUE GET-MAPPED-COMPETENCIES-WS-
CAPABILITY-NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-MAPPED-COMPETENCIES-INTERFACE-NON-FUNCTIONAL-PROPERTIES (NON-
FUNCTIONAL-PROPERTIES)
  NIL)
(DEF-CLASS GET-MAPPED-COMPETENCIES-INTERFACE-CHOREOGRAPHY (CHOREOGRAPHY)
  ((HAS-GROUNDING :VALUE ((GROUNDED-TO-LISP
                           (NORMAL
                            find-matching-generalization-lo))))))

(DEF-CLASS GET-MAPPED-COMPETENCIES-INTERFACE-ORCHESTRATION-PROBLEM-SOLVING-
PATTERN (PROBLEM-SOLVING-PATTERN)
  NIL)

(DEF-CLASS GET-MAPPED-COMPETENCIES-INTERFACE-ORCHESTRATION (ORCHESTRATION)
  ((HAS-PROBLEM-SOLVING-PATTERN :VALUE GET-MAPPED-COMPETENCIES-INTERFACE-
ORCHESTRATION-PROBLEM-SOLVING-PATTERN)))

(DEF-CLASS GET-MAPPED-COMPETENCIES-INTERFACE (INTERFACE) ?INTERFACE
  ((HAS-CHOREOGRAPHY :VALUE GET-MAPPED-COMPETENCIES-INTERFACE-CHOREOGRAPHY)
   (HAS-ORCHESTRATION :VALUE GET-MAPPED-COMPETENCIES-INTERFACE-ORCHESTRATION)
   (HAS-NON-FUNCTIONAL-PROPERTIES :VALUE GET-MAPPED-COMPETENCIES-INTERFACE-
NON-FUNCTIONAL-PROPERTIES)))

(DEF-CLASS GET-MAPPED-COMPETENCIES-WS-PUBLISHER-INFORMATION
(PUBLISHER-INFORMATION)
  ((HAS-ASSOCIATED-WEB-SERVICE-INTERFACE
   :VALUE
   GET-MAPPED-COMPETENCIES-INTERFACE)
   (HAS-WEB-SERVICE-HOST :VALUE "137.108.64.14")
   (HAS-WEB-SERVICE-PORT :VALUE 3001)
   (HAS-WEB-SERVICE-LOCATION :VALUE "/soap")))

```

8.4 Negotiation Layer WSDL

```

<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions
targetNamespace="http://service.soa.negotiation.luisa.atosorigin.es"
xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://service.soa.negotiation.luisa.atosorigin.es"
xmlns:intf="http://service.soa.negotiation.luisa.atosorigin.es"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:tns2="http://message.soa.negotiation.luisa.atosorigin.es"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <!--WSDL created by Apache Axis version: 1.4
Built on Apr 22, 2006 (06:55:48 PDT)-->
  <wsdl:types>
    <schema
targetNamespace="http://message.soa.negotiation.luisa.atosorigin.es"
xmlns="http://www.w3.org/2001/XMLSchema">
      <import

```

```
namespace="http://service.soa.negotiation.luisa.atosorigin.es"/>
    <import
namespace="http://schemas.xmlsoap.org/soap/encoding/" />
    <complexType name="CompetencyDefinition">
        <sequence>
            <element name="competencyCurrentLevel"
nillable="true" type="soapenc:int"/>
            <element name="competencyDescription"
nillable="true" type="soapenc:string"/>
            <element name="competencyId" nillable="true"
type="soapenc:string"/>
            <element name="competencyName" nillable="true"
type="soapenc:string"/>
            <element name="competencyRequiredLevel"
nillable="true" type="soapenc:int"/>
            <element name="competencyType" nillable="true"
type="soapenc:string"/>
        </sequence>
    </complexType>
    <complexType name="EADSIndustryPreferences">
        <sequence>
            <element name="agreement" nillable="true"
type="soapenc:int"/>
            <element name="cost" nillable="true"
type="soapenc:int"/>
            <element name="language" nillable="true"
type="soapenc:string"/>
            <element name="location" nillable="true"
type="soapenc:string"/>
        </sequence>
    </complexType>
    <complexType name="EADSUserPreferences">
        <sequence>
            <element name="alternativeLanguage"
nillable="true" type="soapenc:string"/>
            <element name="alternativeLocation"
nillable="true" type="soapenc:string"/>
            <element name="dif_Plan" nillable="true"
type="soapenc:string"/>
            <element name="durationMeasure" nillable="true"
type="soapenc:string"/>
            <element name="durationOperator"
nillable="true" type="soapenc:string"/>
            <element name="durationValue" nillable="true"
type="soapenc:int"/>
            <element name="group" nillable="true"
type="soapenc:string"/>
            <element name="method" nillable="true"
type="soapenc:string"/>
            <element name="nativeLanguage" nillable="true"
type="soapenc:string"/>
            <element name="position" nillable="true"
type="soapenc:string"/>
            <element name="workingDiscipline"
nillable="true" type="soapenc:string"/>
            <element name="workingLocation" nillable="true"
type="soapenc:string"/>
            <element name="geoLocatizationWorking"
nillable="true" type="soapenc:string"/>
```

```
        </sequence>
    </complexType>
    <complexType name="ProfileEADSReturn">
        <sequence>
            <element name="EADSGap" nillable="true"
type="impl:ArrayOf_tns2_CompetencyDefinition"/>
            <element name="EADSIndustryPreferences"
nillable="true" type="tns2:EADSIndustryPreferences"/>
            <element name="EADSUserPreferences"
nillable="true" type="tns2:EADSUserPreferences"/>
            <element name="EADSProfession" nillable="true"
type="impl:ArrayOf_soapenc_string"/>
            <element name="correlationIdEADS"
nillable="true" type="soapenc:string"/>
        </sequence>
    </complexType>
    <complexType name="Explanation">
        <sequence>
            <element name="comp_id" nillable="true"
type="soapenc:string"/>
            <element name="correlation_id" nillable="true"
type="soapenc:string"/>
            <element name="date" nillable="true"
type="xsd:date"/>
            <element name="lo_id" nillable="true"
type="soapenc:string"/>
            <element name="message" nillable="true"
type="soapenc:string"/>
        </sequence>
    </complexType>
    <complexType name="UHPUniversityPreferences">
        <sequence>
            <element name="cost" nillable="true"
type="soapenc:int"/>
            <element name="language" nillable="true"
type="soapenc:string"/>
        </sequence>
    </complexType>
    <complexType name="UHPUserPreferences">
        <sequence>
            <element name="discipline" nillable="true"
type="soapenc:string"/>
            <element name="duration" nillable="true"
type="soapenc:int"/>
            <element name="durationMeasure" nillable="true"
type="soapenc:string"/>
            <element name="email" nillable="true"
type="soapenc:string"/>
            <element name="os" nillable="true"
type="impl:ArrayOf_soapenc_string"/>
            <element name="sw" nillable="true"
type="impl:ArrayOf_soapenc_string"/>
            <element name="crossSearch" nillable="true"
type="soapenc:int"/>
            <element name="socialRating" nillable="true"
type="soapenc:int"/>
            <element name="tutor" nillable="true"
type="soapenc:int"/>
        </sequence>
    </complexType>
</sequence>
```

```

        </complexType>
        <complexType name="ProfileUHPReturn">
            <sequence>
                <element name="UHPGap" nillable="true"
type="impl:ArrayOf_tns2_CompetyencyDefinition"/>
                <element name="UHPUniversityPreferences"
nillable="true" type="tns2:UHPUniversityPreferences"/>
                <element name="UHPUserPreferences"
nillable="true" type="tns2:UHPUserPreferences"/>
                <element name="correlationIdUHP"
nillable="true" type="soapenc:string"/>
            </sequence>
        </complexType>
        <complexType name="CompetencyLevel">
            <sequence>
                <element name="name" nillable="true"
type="soapenc:string"/>
                <element name="requiredLevel" type="xsd:int"/>
            </sequence>
        </complexType>
        <complexType name="LOResultsSOA">
            <sequence>
                <element name="competency" nillable="true"
type="impl:ArrayOf_soapenc_string"/>
                <element name="learningObjectComposed"
nillable="true" type="impl:ArrayOf_soapenc_string"/>
                <element name="learningObjectCost"
nillable="true" type="soapenc:double"/>
                <element name="learningObjectDescription"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectDurationUnit"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectDurationValue"
nillable="true" type="soapenc:int"/>
                <element name="learningObjectId"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectLanguage"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectRanking"
nillable="true" type="soapenc:double"/>
                <element name="learningObjectTitle"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectURL"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectGeoLocation"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectDRMEvaluation"
nillable="true" type="soapenc:int"/>
                <element name="learningObjectDRMExplanation"
nillable="true" type="soapenc:string"/>
                <element name="learningObjectLearningStyle"
nillable="true" type="soapenc:string"/>
                <element
name="learningObjectCollaborativeFilteringRating" nillable="true"
type="soapenc:int"/>
                <element name="learningObjectResourceType"
nillable="true" type="soapenc:string"/>
            </sequence>
        </complexType>

```

```

        <complexType name="CompetencyResults">
            <sequence>
                <element name="learningObjectInfo"
nillable="true" type="tns2:LOResultsSOA"/>
            </sequence>
        </complexType>
        <complexType name="GetLOsReturn">
            <sequence>
                <element name="alternativePlan" nillable="true"
type="impl:ArrayOf_tns2_CompetencyResults"/>
                <element name="correlationId" nillable="true"
type="soapenc:string"/>
                <element name="workPlan" nillable="true"
type="impl:ArrayOf_tns2_CompetencyResults"/>
            </sequence>
        </complexType>
        <complexType name="UHPCompositionLO">
            <sequence>
                <element name="LOId" nillable="true"
type="soapenc:string"/>
                <element name="learningStyle" nillable="true"
type="soapenc:string"/>
                <element name="language" nillable="true"
type="soapenc:string"/>
            </sequence>
        </complexType>
        <complexType name="UHPCompositionRequest">
            <sequence>
                <element name="LOIds" nillable="true"
type="impl:ArrayOf_UHPCompositionLO"/>
                <element name="numberOfResourcesMin"
nillable="true" type="soapenc:int"/>
                <element name="numberOfResourcesMax"
nillable="true" type="soapenc:int"/>
                <element name="numberOfResourcesLessons"
nillable="true" type="soapenc:int"/>
                <element name="numberOfResourcesExercises"
nillable="true" type="soapenc:int"/>
                <element name="numberOfResourcesExams"
nillable="true" type="soapenc:int"/>
                <element name="consistencyInRepository"
nillable="true" type="soapenc:int"/>
                <element name="consistencyInLanguage"
nillable="true" type="soapenc:int"/>
                <element name="learningStyle" nillable="true"
type="soapenc:int"/>
            </sequence>
        </complexType>
        <complexType name="UHPCompositionReturn">
            <sequence>
                <element name="compositionAlternatives"
nillable="true" type="impl:ArrayOf_tns2_CompositionAlternatives"/>
                <element name="correlationIdUHP"
nillable="true" type="soapenc:string"/>
            </sequence>
        </complexType>
        <complexType name="UHPSaveCompositionRequest">
            <sequence>
                <element name="LOIds" nillable="true"

```

```

type="impl:ArrayOf_soapenc_string" />
    </sequence>
</complexType>
<complexType name="UHPPackagesSavedForTutor">
    <sequence>
        <element name="packagesSaved" nillable="true"
type="impl:ArrayOf_tns2_UHPPackageSaved" />
    </sequence>
</complexType>
<complexType name="UHPPackageSaved">
    <sequence>
        <element name="LOId" nillable="true"
type="soapenc:string" />
        <element name="package" nillable="true"
type="impl:ArrayOf_soapenc_string" />
    </sequence>
</complexType>
</schema>
<schema
targetNamespace="http://service.soa.negotiation.luisa.atosorigin.es"
xmlns="http://www.w3.org/2001/XMLSchema">
    <import
namespace="http://message.soa.negotiation.luisa.atosorigin.es" />
    <import
namespace="http://schemas.xmlsoap.org/soap/encoding/" />
    <complexType name="ArrayOf_tns2_CompetyDefinition">
        <complexContent>
            <restriction base="soapenc:Array">
                <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns2:CompetyDefinition[]" />
            </restriction>
        </complexContent>
    </complexType>
    <complexType name="ArrayOf_soapenc_string">
        <complexContent>
            <restriction base="soapenc:Array">
                <attribute ref="soapenc:arrayType"
wsdl:arrayType="soapenc:string[]" />
            </restriction>
        </complexContent>
    </complexType>
    <complexType name="MsgFault">
        <sequence>
            <element name="negotiationFaultActor"
nillable="true" type="soapenc:string" />
            <element name="negotiationFaultCode"
nillable="true" type="soapenc:string" />
            <element name="negotiationFaultDescription"
nillable="true" type="soapenc:string" />
        </sequence>
    </complexType>
    <complexType name="ArrayOf_UHPCompositionLO">
        <complexContent>
            <restriction base="soapenc:Array">
                <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns2:UHPCompositionLO[]" />
            </restriction>
        </complexContent>
    </complexType>

```

```

        <complexType name="ArrayOf_tns2_Explanation">
            <complexContent>
                <restriction base="soapenc:Array">
                    <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns2:Explanation[]" />
                </restriction>
            </complexContent>
        </complexType>
        <complexType name="ArrayOf_tns2_CompetyencyLevel">
            <complexContent>
                <restriction base="soapenc:Array">
                    <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns2:CompetyencyLevel[]" />
                </restriction>
            </complexContent>
        </complexType>
        <complexType name="ArrayOf_tns2_CompetyencyResults">
            <complexContent>
                <restriction base="soapenc:Array">
                    <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns2:CompetyencyResults[]" />
                </restriction>
            </complexContent>
        </complexType>
        <complexType name="ArrayOf_tns2_CompositionAlternatives">
            <complexContent>
                <restriction base="soapenc:Array">
                    <attribute ref="soapenc:arrayType"
wsdl:arrayType="impl:ArrayOf_soapenc_string[]" />
                </restriction>
            </complexContent>
        </complexType>
        <complexType name="ArrayOf_tns2_UHPPackageSaved">
            <complexContent>
                <restriction base="soapenc:Array">
                    <attribute ref="soapenc:arrayType"
wsdl:arrayType="tns2:UHPPackageSaved[]" />
                </restriction>
            </complexContent>
        </complexType>
    </schema>
</wsdl:types>
<wsdl:message name="getProfileUHPRequest">
    <wsdl:part name="userId" type="soapenc:string" />
</wsdl:message>
<wsdl:message name="getProfileEADSResponse">
    <wsdl:part name="getProfileEADSReturn"
type="tns2:ProfileEADSReturn" />
</wsdl:message>
<wsdl:message name="getLOsForCompetenciesUHPRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string" />
    <wsdl:part name="comp_level"
type="impl:ArrayOf_tns2_CompetyencyLevel" />
</wsdl:message>
<wsdl:message name="deleteProfileUHPResponse">
    <wsdl:part name="deleteProfileUHPReturn" type="xsd:int" />
</wsdl:message>
<wsdl:message name="getExplanationResponse">
    <wsdl:part name="getExplanationReturn"

```

```
type="impl:ArrayOf_tns2_Explanation"/>
  </wsdl:message>
  <wsdl:message name="getLOsForCompetenciesEADSResponse">
    <wsdl:part name="getLOsForCompetenciesEADSReturn"
type="tns2:GetLOsReturn"/>
  </wsdl:message>
  <wsdl:message name="deleteProfileUHPRequest">
    <wsdl:part name="userId" type="soapenc:string"/>
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  </wsdl:message>
  <wsdl:message name="getLOsForCompetenciesEADSRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
    <wsdl:part name="comp_level"
type="impl:ArrayOf_tns2_CompetencyLevel"/>
  </wsdl:message>
  <wsdl:message name="saveProfileEADSResponse">
    <wsdl:part name="saveProfileEADSReturn" type="xsd:int"/>
  </wsdl:message>
  <wsdl:message name="getLOsForCompetenciesUHPResponse">
    <wsdl:part name="getLOsForCompetenciesUHPReturn"
type="tns2:GetLOsReturn"/>
  </wsdl:message>
  <wsdl:message name="saveUniversityPreferencesUHPRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
    <wsdl:part name="preferences"
type="tns2:UHPUniversityPreferences"/>
  </wsdl:message>
  <wsdl:message name="MsgFault">
    <wsdl:part name="fault" type="impl:MsgFault"/>
  </wsdl:message>
  <wsdl:message name="getLOsByMetadataEADSRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
    <wsdl:part name="metadata" type="soapenc:string"/>
    <wsdl:part name="values" type="impl:ArrayOf_soapenc_string"/>
  </wsdl:message>
  <wsdl:message name="getLOsForCompetenciesOtherProfessionEADSResponse">
    <wsdl:part name="getLOsForCompetenciesOtherProfessionEADSReturn"
type="tns2:GetLOsReturn"/>
  </wsdl:message>
  <wsdl:message name="getLOsForCompetenciesOtherProfessionEADSRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
    <wsdl:part name="comp" type="impl:ArrayOf_soapenc_string"/>
  </wsdl:message>
  <wsdl:message name="savePreferencesEADSRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
    <wsdl:part name="preferences" type="tns2:EADSUserPreferences"/>
  </wsdl:message>
  <wsdl:message name="getProfileUHPResponse">
    <wsdl:part name="getProfileUHPReturn"
type="tns2:ProfileUHPReturn"/>
  </wsdl:message>
  <wsdl:message name="getExplanationRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  </wsdl:message>
  <wsdl:message name="getLOsForCompetenciesEADSUsingStrategiesRequest">
    <wsdl:part name="correlationIdIn" type="soapenc:string"/>
    <wsdl:part name="strategy" type="soapenc:string"/>
  </wsdl:message>
  <wsdl:message name="savePreferencesUHPRequest">
```

```
<wsdl:part name="correlationIdIn" type="soapenc:string"/>
  <wsdl:part name="preferences" type="tns2:UHPUserPreferences"/>
</wsdl:message>
<wsdl:message name="getLOsByMetadataEADSResponse">
  <wsdl:part name="getLOsByMetadataEADSReturn"
type="tns2:GetLOsReturn"/>
</wsdl:message>
<wsdl:message name="saveProfileUHPRequest">
  <wsdl:part name="userId" type="soapenc:string"/>
  <wsdl:part name="profile" type="tns2:ProfileUHPReturn"/>
</wsdl:message>
<wsdl:message name="getCOsByProfessionRequest">
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  <wsdl:part name="profession" type="soapenc:string"/>
</wsdl:message>
<wsdl:message name="saveProfileUHPResponse">
  <wsdl:part name="saveProfileUHPReturn" type="xsd:int"/>
</wsdl:message>
<wsdl:message name="deleteProfileEADSResponse">
  <wsdl:part name="deleteProfileEADSReturn" type="xsd:int"/>
</wsdl:message>
<wsdl:message name="saveUniversityPreferencesUHPResponse">
  <wsdl:part name="saveUniversityPreferencesUHPReturn"
type="xsd:int"/>
</wsdl:message>
<wsdl:message name="saveIndustryPreferencesEADSRequest">
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  <wsdl:part name="preferences"
type="tns2:EADSIndustryPreferences"/>
</wsdl:message>
<wsdl:message name="getLOsForCompetenciesEADSUsingStrategiesResponse">
  <wsdl:part name="getLOsForCompetenciesEADSUsingStrategiesReturn"
type="tns2:GetLOsReturn"/>
</wsdl:message>
<wsdl:message name="savePreferencesUHPResponse">
  <wsdl:part name="savePreferencesUHPReturn" type="xsd:int"/>
</wsdl:message>
<wsdl:message name="getProfileEADSRequest">
  <wsdl:part name="userId" type="soapenc:string"/>
</wsdl:message>
<wsdl:message name="saveIndustryPreferencesEADSResponse">
  <wsdl:part name="saveIndustryPreferencesEADSReturn"
type="xsd:int"/>
</wsdl:message>
<wsdl:message name="saveProfileEADSRequest">
  <wsdl:part name="userId" type="soapenc:string"/>
  <wsdl:part name="profile" type="tns2:ProfileEADSReturn"/>
</wsdl:message>
<wsdl:message name="getCOsByProfessionResponse">
  <wsdl:part name="getCOsByProfessionReturn"
type="impl:ArrayOf_tns2_CompetencyLevel"/>
</wsdl:message>
<wsdl:message name="deleteProfileEADSRequest">
  <wsdl:part name="userId" type="soapenc:string"/>
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
</wsdl:message>
<wsdl:message name="savePreferencesEADSResponse">
  <wsdl:part name="savePreferencesEADSReturn" type="xsd:int"/>
</wsdl:message>
```

```

<wsdl:message name="evaluateCorrelationIdRequest">
  <wsdl:part name="correlationIdIn" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="evaluateCorrelationIdResponse">
  <wsdl:part name="exists" type="xsd:int"/>
</wsdl:message>
<wsdl:message name="getTentativeCompositionRequest">
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  <wsdl:part name="compositionRequest"
type="tns2:UHPCompositionRequest"/>
</wsdl:message>
<wsdl:message name="getTentativeCompositionResponse">
  <wsdl:part name="compositionReturn"
type="tns2:UHPCompositionReturn"/>
</wsdl:message>
<wsdl:message name="saveCompositionForTutorRequest">
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  <wsdl:part name="saveCompositionRequest"
type="tns2:UHPSaveCompositionRequest"/>
</wsdl:message>
<wsdl:message name="saveCompositionForTutorResponse">
  <wsdl:part name="ComposedLOId" type="xsd:string"/>
</wsdl:message>
<wsdl:message name="packagesSavedForTutorRequest">
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
</wsdl:message>
<wsdl:message name="packagesSavedForTutorResponse">
  <wsdl:part name="packagesSavedForTutorReturn"
type="tns2:UHPPackagesSavedForTutor"/>
</wsdl:message>
<wsdl:message name="deletePackageRequest">
  <wsdl:part name="correlationIdIn" type="soapenc:string"/>
  <wsdl:part name="LOId" type="soapenc:string"/>
</wsdl:message>
<wsdl:message name="deletePackageResponse">
  <wsdl:part name="ack" type="xsd:int"/>
</wsdl:message>
<wsdl:portType name="NegotiationLayerWS">
  <wsdl:operation name="getProfileEADS" parameterOrder="userId">
    <wsdl:input message="impl:getProfileEADSRequest"
name="getProfileEADSRequest"/>
    <wsdl:output message="impl:getProfileEADSResponse"
name="getProfileEADSResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
  </wsdl:operation>
  <wsdl:operation name="getExplanation"
parameterOrder="correlationIdIn">
    <wsdl:input message="impl:getExplanationRequest"
name="getExplanationRequest"/>
    <wsdl:output message="impl:getExplanationResponse"
name="getExplanationResponse"/>
  </wsdl:operation>
  <wsdl:operation name="deleteProfileUHP" parameterOrder="userId
correlationIdIn">
    <wsdl:input message="impl:deleteProfileUHPRequest"
name="deleteProfileUHPRequest"/>
    <wsdl:output message="impl:deleteProfileUHPResponse"
name="deleteProfileUHPResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
  </wsdl:operation>
</wsdl:portType>

```

```

        </wsdl:operation>
        <wsdl:operation name="deleteProfileEADS" parameterOrder="userId
correlationIdIn">
            <wsdl:input message="impl:deleteProfileEADSRequest"
name="deleteProfileEADSRequest" />
            <wsdl:output message="impl:deleteProfileEADSResponse"
name="deleteProfileEADSResponse" />
            <wsdl:fault message="impl:MsgFault" name="MsgFault" />
        </wsdl:operation>
        <wsdl:operation name="getProfileUHP" parameterOrder="userId">
            <wsdl:input message="impl:getProfileUHPRequest"
name="getProfileUHPRequest" />
            <wsdl:output message="impl:getProfileUHPResponse"
name="getProfileUHPResponse" />
            <wsdl:fault message="impl:MsgFault" name="MsgFault" />
        </wsdl:operation>
        <wsdl:operation name="savePreferencesEADS"
parameterOrder="correlationIdIn preferences">
            <wsdl:input message="impl:savePreferencesEADSRequest"
name="savePreferencesEADSRequest" />
            <wsdl:output message="impl:savePreferencesEADSResponse"
name="savePreferencesEADSResponse" />
        </wsdl:operation>
        <wsdl:operation name="savePreferencesUHP"
parameterOrder="correlationIdIn preferences">
            <wsdl:input message="impl:savePreferencesUHPRequest"
name="savePreferencesUHPRequest" />
            <wsdl:output message="impl:savePreferencesUHPResponse"
name="savePreferencesUHPResponse" />
        </wsdl:operation>
        <wsdl:operation name="saveProfileEADS" parameterOrder="userId
profile">
            <wsdl:input message="impl:saveProfileEADSRequest"
name="saveProfileEADSRequest" />
            <wsdl:output message="impl:saveProfileEADSResponse"
name="saveProfileEADSResponse" />
            <wsdl:fault message="impl:MsgFault" name="MsgFault" />
        </wsdl:operation>
        <wsdl:operation name="saveProfileUHP" parameterOrder="userId
profile">
            <wsdl:input message="impl:saveProfileUHPRequest"
name="saveProfileUHPRequest" />
            <wsdl:output message="impl:saveProfileUHPResponse"
name="saveProfileUHPResponse" />
            <wsdl:fault message="impl:MsgFault" name="MsgFault" />
        </wsdl:operation>
        <wsdl:operation name="saveIndustryPreferencesEADS"
parameterOrder="correlationIdIn preferences">
            <wsdl:input
message="impl:saveIndustryPreferencesEADSRequest"
name="saveIndustryPreferencesEADSRequest" />
            <wsdl:output
message="impl:saveIndustryPreferencesEADSResponse"
name="saveIndustryPreferencesEADSResponse" />
        </wsdl:operation>
        <wsdl:operation name="saveUniversityPreferencesUHP"
parameterOrder="correlationIdIn preferences">
            <wsdl:input
message="impl:saveUniversityPreferencesUHPRequest"

```

```
name="saveUniversityPreferencesUHPRequest" />
    <wsdl:output
message="impl:saveUniversityPreferencesUHPResponse"
name="saveUniversityPreferencesUHPResponse" />
    </wsdl:operation>
    <wsdl:operation name="getCOsByProfession"
parameterOrder="correlationIdIn profession">
    <wsdl:input message="impl:getCOsByProfessionRequest"
name="getCOsByProfessionRequest" />
    <wsdl:output message="impl:getCOsByProfessionResponse"
name="getCOsByProfessionResponse" />
    </wsdl:operation>
    <wsdl:operation name="getLOsForCompetenciesOtherProfessionEADS"
parameterOrder="correlationIdIn comp">
    <wsdl:input
message="impl:getLOsForCompetenciesOtherProfessionEADSRequest"
name="getLOsForCompetenciesOtherProfessionEADSRequest" />
    <wsdl:output
message="impl:getLOsForCompetenciesOtherProfessionEADSResponse"
name="getLOsForCompetenciesOtherProfessionEADSResponse" />
    <wsdl:fault message="impl:MsgFault" name="MsgFault" />
    </wsdl:operation>
    <wsdl:operation name="getLOsByMetadataEADS"
parameterOrder="correlationIdIn metadata values">
    <wsdl:input message="impl:getLOsByMetadataEADSRequest"
name="getLOsByMetadataEADSRequest" />
    <wsdl:output message="impl:getLOsByMetadataEADSResponse"
name="getLOsByMetadataEADSResponse" />
    <wsdl:fault message="impl:MsgFault" name="MsgFault" />
    </wsdl:operation>
    <wsdl:operation name="getLOsForCompetenciesEADSUsingStrategies"
parameterOrder="correlationIdIn strategy">
    <wsdl:input
message="impl:getLOsForCompetenciesEADSUsingStrategiesRequest"
name="getLOsForCompetenciesEADSUsingStrategiesRequest" />
    <wsdl:output
message="impl:getLOsForCompetenciesEADSUsingStrategiesResponse"
name="getLOsForCompetenciesEADSUsingStrategiesResponse" />
    <wsdl:fault message="impl:MsgFault" name="MsgFault" />
    </wsdl:operation>
    <wsdl:operation name="getLOsForCompetenciesEADS"
parameterOrder="correlationIdIn comp_level">
    <wsdl:input message="impl:getLOsForCompetenciesEADSRequest"
name="getLOsForCompetenciesEADSRequest" />
    <wsdl:output
message="impl:getLOsForCompetenciesEADSResponse"
name="getLOsForCompetenciesEADSResponse" />
    <wsdl:fault message="impl:MsgFault" name="MsgFault" />
    </wsdl:operation>
    <wsdl:operation name="getLOsForCompetenciesUHP"
parameterOrder="correlationIdIn comp_level">
    <wsdl:input message="impl:getLOsForCompetenciesUHPRequest"
name="getLOsForCompetenciesUHPRequest" />
    <wsdl:output
message="impl:getLOsForCompetenciesUHPResponse"
name="getLOsForCompetenciesUHPResponse" />
    <wsdl:fault message="impl:MsgFault" name="MsgFault" />
    </wsdl:operation>
    <wsdl:operation name="evaluateCorrelationId"
```

```

parameterOrder="correlationIdIn">
    <wsdl:input message="impl:evaluateCorrelationIdRequest"
name="evaluateCorrelationIdRequest"/>
    <wsdl:output message="impl:evaluateCorrelationIdResponse"
name="evaluateCorrelationIdResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
</wsdl:operation>
<wsdl:operation name="getTentativeComposition"
parameterOrder="correlationIdIn compositionRequest">
    <wsdl:input message="impl:getTentativeCompositionRequest"
name="getTentativeCompositionRequest"/>
    <wsdl:output message="impl:getTentativeCompositionResponse"
name="getTentativeCompositionResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
</wsdl:operation>
<wsdl:operation name="saveCompositionForTutor"
parameterOrder="correlationIdIn saveCompositionRequest">
    <wsdl:input message="impl:saveCompositionForTutorRequest"
name="saveCompositionForTutorRequest"/>
    <wsdl:output message="impl:saveCompositionForTutorResponse"
name="saveCompositionForTutorResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
</wsdl:operation>
<wsdl:operation name="packagesSavedForTutor"
parameterOrder="correlationIdIn">
    <wsdl:input message="impl:packagesSavedForTutorRequest"
name="packagesSavedForTutorRequest"/>
    <wsdl:output message="impl:packagesSavedForTutorResponse"
name="packagesSavedForTutorResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
</wsdl:operation>
<wsdl:operation name="deletePackage"
parameterOrder="correlationIdIn LOId">
    <wsdl:input message="impl:deletePackageRequest"
name="deletePackageRequest"/>
    <wsdl:output message="impl:deletePackageResponse"
name="deletePackageResponse"/>
    <wsdl:fault message="impl:MsgFault" name="MsgFault"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="NegotiationLayerWSSoapBinding"
type="impl:NegotiationLayerWS">
    <wsdlsoap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="getProfileEADS">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="getProfileEADSRequest">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:input>
        <wsdl:output name="getProfileEADSResponse">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:output>
        <wsdl:fault name="MsgFault">

```

```

        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="getExplanation">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="getExplanationRequest">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
            </wsdl:input>
            <wsdl:output name="getExplanationResponse">
                <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
                </wsdl:output>
            </wsdl:operation>
            <wsdl:operation name="deleteProfileUHP">
                <wsdlsoap:operation soapAction=""/>
                <wsdl:input name="deleteProfileUHPRequest">
                    <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
                    </wsdl:input>
                    <wsdl:output name="deleteProfileUHPResponse">
                        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
                        </wsdl:output>
                    <wsdl:fault name="MsgFault">
                        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
                        </wsdl:fault>
                    </wsdl:operation>
                    <wsdl:operation name="deleteProfileEADS">
                        <wsdlsoap:operation soapAction=""/>
                        <wsdl:input name="deleteProfileEADSRequest">
                            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
                            </wsdl:input>
                            <wsdl:output name="deleteProfileEADSResponse">
                                <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
                                </wsdl:output>
                            <wsdl:fault name="MsgFault">
                                <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"

```

```
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="getProfileUHP">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="getProfileUHPRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="getProfileUHPResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
    <wsdl:fault name="MsgFault">
        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="savePreferencesEADS">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="savePreferencesEADSRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="savePreferencesEADSResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="savePreferencesUHP">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="savePreferencesUHPRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="savePreferencesUHPResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="saveProfileEADS">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="saveProfileEADSRequest">
        <wsdlsoap:body
```

```
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="saveProfileEADSResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
    <wsdl:fault name="MsgFault">
        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="saveProfileUHP">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="saveProfileUHPRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="saveProfileUHPResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
    <wsdl:fault name="MsgFault">
        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="saveIndustryPreferencesEADS">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="saveIndustryPreferencesEADSRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="saveIndustryPreferencesEADSResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="saveUniversityPreferencesUHP">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="saveUniversityPreferencesUHPRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="saveUniversityPreferencesUHPResponse">
        <wsdlsoap:body
```



```
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="getLOsForCompetenciesUHPResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
    <wsdl:fault name="MsgFault">
        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="evaluateCorrelationId">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="evaluateCorrelationIdRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="evaluateCorrelationIdResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
    <wsdl:fault name="MsgFault">
        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="getTentativeComposition">
    <wsdlsoap:operation soapAction=""/>
    <wsdl:input name="getTentativeCompositionRequest">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:input>
    <wsdl:output name="getTentativeCompositionResponse">
        <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:output>
    <wsdl:fault name="MsgFault">
        <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="saveCompositionForTutor">
    <wsdlsoap:operation soapAction=""/>
```

```
        <wsdl:input name="saveCompositionForTutorRequest">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:input>
        <wsdl:output name="saveCompositionForTutorResponse">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:output>
        <wsdl:fault name="MsgFault">
            <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="packagesSavedForTutor">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="packagesSavedForTutorRequest">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:input>
        <wsdl:output name="packagesSavedForTutorResponse">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:output>
        <wsdl:fault name="MsgFault">
            <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="deletePackage">
        <wsdlsoap:operation soapAction=""/>
        <wsdl:input name="deletePackageRequest">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:input>
        <wsdl:output name="deletePackageResponse">
            <wsdlsoap:body
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:output>
        <wsdl:fault name="MsgFault">
            <wsdlsoap:fault
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" name="MsgFault"
namespace="http://service.soa.negotiation.luisa.atosorigin.es"
use="encoded"/>
        </wsdl:fault>
    </wsdl:operation>
```

```
        </wsdl:fault>
    </wsdl:operation>
</wsdl:binding>
<wsdl:service name="NegotiationLayerWSService">
    <wsdl:port binding="impl:NegotiationLayerWSSoapBinding"
name="NegotiationLayerWS">
        <wsdlsoap:address location="
http://localhost:8080/NegotiationLayerService2/services/NegotiationLayerWS"/>
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

8.5 Query Resolver Java Interface

```
public interface QueryResolverIfc {
    public RequestBO setUpQuery(CompetencyLevel levels, ProfileIfc objProfile);
    public CompetencyLevel[] setupInputsRule(ProfileIfc objProfile,
CompetencyLevel[] comp);
    public Collection<LOParserAbstract> filterRules(CompetencyLevel levels,
Collection<LOParserAbstract> cLearningObjects);
    public RequestBO refineQueryRules(ProfileIfc profile,String sCorrelationId,
Collection cLearningObjects, RequestBO objRequestBO) throws
NegotiationLayerException;
    public List<CompetencyResultsList> rankingRules(Collection
<CompetencyTempList> cLearningObjects, ProfileIfc objProfile) throws
NegotiationLayerException;
    public Collection<LOParserAbstract> parseXML(String xml) throws
NegotiationLayerException;
    public ProfileIfc parseXMLProfile(String xml) throws
NegotiationLayerException;
}
```