Description and composition of services for distant learning and research activities

Oussama Kassem Zein, Yvon Kermarrec

Description and composition of services for distant learning and research activities

Oussama Kassem Zein and Yvon Kermarrec
ENST Bretagne, Dept. LUSSI
Technopôle Brest Iroise, BP 832, 29285 Brest Cedex, France
Email : {Oussama.Zein,Yvon.Kermarrec}@enst-bretagne.fr
Tel : + 33 2 29 00 12 85 Fax : + 33 2 29 00 10 30

Abstract

In this paper, we present our approach to describe and compose services for distant learning and research activities. In this context, we propose a metadata model for indexing services with three dimensions: it can be viewed as learning resources, as services that contribute and help researchers to perform their tasks in a community and as services with a large scope. This model can be used by clients (learners/teachers or researchers) to query and discover services via for example a trader we have developed. We extend the functionality of the trader to be a composition engine enabling learners/researchers to compose several services or resources that can help them in their distant learning and research activities. We use the documents (databases, web documents, etc.) as an application for our approach of service description. For this purpose, we have decided to add an interface to the document. This allows documents to be reused and shared by using the approaches defined for web services.

KEYWORDS: composition of services, service description, service discovery, service composition.

1 Introduction and Context

Many different approaches for describing, managing and providing learning resources have been developed over the years to accomodate the massive flow of information of the internet like Dublin Core [1] and LOM (Learning Object Metadata) [2]. Nevertheless, a clear understanding and consensus about what constitutes a learning resource has not yet been reached. Even though, we can find analogies with software engineering.

These approaches provide a set of properties that characterize a learning resource. These properties can be used by clients (learners/teachers) to discover e-learning services via UDDI [3] or a trader we have developed [4].

In this paper, we propose to extend the nature of a learning resource to be a service and to make it accessible via through an interface, as it is the case for a learning service. In this context, our proposal is to make a learning resource not only as a downloadable thing- that is only executed on the client machine with no interaction with servers with during execution time- but also as a service accessible by an interface. We define a metadata model for learning services description which can be stored in UDDI or in our trader [4]. We have used ontologies [5] to index and store this model as for our previous R&D activities. An ontology is a consensus between clients and service providers, so they can cooperate and share a common vocabulary. Clients and providers that can be teachers or students can use this model to query and publish learning services through ontologies.

We extend the metadata model to make a client not only a learner or a teacher but also a researcher (a PhD student, a professor, etc.) to do distant research. In this context, a researcher can publish his services (software, tools, simulation results, etc.) that can contribute and help other researchers in their research. So, how can we describe these services to make possible the distant research between researchers?

In a previous work [6], we have extended the functionalities of our trader to be a composition engine that allows combining several services into one service based on their behavior. We have proposed three models of composition: static, semi-dynamic and dynamic. Based on these approaches and the metadata model we propose to describe e-learning services, we present in this paper how clients (learners/teachers or researchers) can benefit from the functionalities of many services to get the appropriate results in their distant learning and research activities and access resources made available by them in the community. We use the documents as an application domain for our approach of description and discovery of services for distant learning and research. We describe a document with a service by adding to it an interface. This allows clients to use the same type of interface to interact with different documents (databases, web documents, etc.) and allows

documents to be reused and shared by using the approaches defined for web services.

This paper is organized as follows. Section 2 presents our proposal of a metadata model that describes and characterizes learning and research services. Section 3 illustrates our proposal of ontology managing interfaces that allow clients and servers to query and advertise learning services based on the underlying use of ontologies. Section 4 presents the service behavior description. Section 5 illustrates our approach allowing the service composition when a service search is unsuccessful. Section 6 presents an application using our approach to describe documents. Section 7 presents our related work. Finally, the conclusion raises issues and presents our future work in section 8.

2 Our Proposed Metadata Model

In this section, we present our metadata model to describe learning services. We propose to describe a learning service with three dimensions: as learning resources, as services that contribute and help researchers to perform their tasks in a community and as general purpose services. This model provides a set of properties that characterizes learning services. In the next sections, we build an ontology which is constituted of concepts to describe and store the learning service properties.

1. **Dimension 1: learning resource**

   We have based our model on Dublin Core [1] and IEEE LOM [2] to ensure compatibility with existing platforms and to allow the description of a learning resource in general. They define a set of elements like:

   - **Title**: the name given to the resource by its creator.
   - **Creator**: The person(s) or organization(s) primarily responsible for the intellectual content of the resource.
   - **Subject**: the topic of the resource, or keywords or phrases that describe the subject or content of the resource.
   - **Description**: a textual description of the content of the resource, including abstracts in the case of document and objects and content descriptions in the case of visual resources.
   - **Publisher**: the person(s) or organization(s) in addition to those specified in the Creator element who have made significant intellectual contributions to the resource but whose is secondary to the individuals or entities specified in the Creator element (for example, editors).
   - **Date**: the date the resource was made available in its present form.
   - **Language**: language(s) of the intellectual content of the resource.
   - **Format**: the data representation of the resource, such as text/html, Postscript file, JPEG image and so on.
   - **Rights**: the content of this element is intended to be a link (a URL for example) to a copyright notice, a rights-management statement and so on.

2. **Dimension 2: research resource**

   Our investigation is to describe services that contribute and help researchers to collaborate, to work together and to exchange the information in their research activities. A service can be a software (a program/application or a component), a document and so on that is provided by a researcher (the author) and can be used by other researchers to help them in their research activities. For this purpose, we must describe the context of the component precisely and we have identified three major fields: the research areas covered by the service, the researchers and the type of service (software, document) provided and so on.

   A research area can be described by:

   - **Indexes**: key words that describe the research area.
   - **References**: a set of papers, documents and URLs that are useful in the research area.
   - **Links**: which are towards other areas and topics which are linked to the research area.
   - **Description**: a short textual description of the research area.

   A researcher (the author of the service) can be described by:

   - **Personal information**: name, email, phone number(s), web site, CV, etc.
   - **Quality and title**: PhD thesis, professor or engineer for example.
   - **Research areas**: the list of the research areas.
   - **Teaching areas**: the list of the teaching areas.
   - **Publications**: references to the published papers.
   - **Department**: to which the researcher is attached.

   These indexes can have default values according to provider profile.

   A department can be described by:

   - **Name**: of the department and the university or organization to which the department is attached.

- **Address**: the location of the department.
- **Description**: a short description of the department.
- **Research areas**: the list of the research orientations of the department.
- **List of researchers** attached to the department.
- **Research groups** constituted in the department.

A service type can be a learning resource, a software, a program, a document and so on.

3. **Dimension 3: service with a large scope**

We have based our proposal on [7], [8], [9], [10] to describe a service with a large scope. We can identify a service by a set of characteristics like: delivery systems (in the case of learning services), payment and pricing, location, quality of service, requesting and delivery channels (PDA, mobile phone, etc.), helping tools (chatting room, mailing list, etc.) and so on.

We present some of these properties:

- **Provider and Location**: the Provider defines the characteristics of the service provider including his name and his address (e.g. the name of his city, street names, postal codes, and country codes, etc.). The Location determines the service location like a company address or an URL address, and so on.
- **Request and delivery channels**: with the introduction of the internet and new communication devices (mobile phones, pagers, etc.), there has been an increase in the number of request and delivery channels available to consumers. A channel is the means by which a user requests a service or receives the resulting output from a service. These are referred to as the request and delivery channels respectively.
- **Payment and pricing**: payment is the business process defined by the service provider for collecting the price of the service from the consumer. It can be conducted in single or multiple stages (i.e. installments), using various mediums (e.g. direct cash exchange, credit or debit card, cheque, direct debit, account, etc.). Payment can be made before delivery, at delivery, after delivery, or any combination of the above.

  Pricing is the charge for the service being provided. It is largely at the discretion of the service provider and as such, we consider a service to have a nominal price.

  Pricing and payment can include the identifier of the entity to which the payment is addressed, which can be different from the identifier of the service provider. They include the payment channel. This is the method used for conducting the payment (e.g. internet, email, post and phone, etc.).
- **Environment** : it determines the characteristics of the environment used to exploit the service like the bandwidth and which many appear as requirements.
- **Delivery system**: like visio-conference in the case of learning resources.
- **Helping tools**: the tools (like chatting room, mailing list and so on) that help clients to use the service.

The clients and service providers of our metadata model can be students (dimensions 1 and 3), teachers (dimensions 1, 2 and 3), researchers (dimensions 2 and 3).

The metadata model provides a complete vision about learning service description since it allows clients and servers to query and advertise services using in learning, research and as general purpose services. This model extends services to be used by researchers/teachers and not only by teachers. It is flexible because it is compatible with existing approaches like IEEE LOM, Dublin Core and DAML-S.

3 Ontology Managing Interfaces

We have used our earlier approach [4] for service discovery based on ontologies. We have selected ontologies because they make it possible to provide a shared consensus that can help share/reuse. In this approach for the ontology managing, we have used a tool named Ontobroker [11] which is a result of a european research program that allows managing ontologies. It consists of a query interface for formulating queries and an inference engine to derive new facts.

We first create an ontology describing the properties of learning services that we have defined in Section 2. We can use the facilities and the structures provided by an ontology like the inheritance relationships to construct the metadata model for our service. If the metadata are more structured by an ontology, the queries can benefit of the ontology organization: they are more flexible and more powerful. By using ontologies, we can benefit from all the power of the logic of first order.

The following example is a concept in the ontology written in F-Logic, to describe the location of the service provider:

```
Provider_address [ city_name =>> STRING;
        street_name =>> STRING;
        postal_code =>> INTEGER;
        country_code =>> INTEGER
        .
```

.
.].

We have implemented the tools that allow us to manage an ontology and we have defined the following interfaces:

- Importing function: it is used by clients. It takes a query written in a logic language (e.g., F-Logic) as the input, it makes a request to Ontobroker for the service required and returns the matching service offers to the client.

- Exporting function: it is used by service providers to index and describe services. It allows a Fact to be added to the ontology. A service offer is an instance of a service. It is described by a fact. To add it to the ontology, a service provider can call this function.

- A function is defined to delete a fact from an ontology.

4 Service behavior description

SDL [12] and Interface Automata [13] have been designed to specify and to describe the functional behavior of telecommunication systems. It describes a process behavior by an automaton which can then be exploited to validate the specification. Therefore, based on SDL and Interface Automata, we describe the service behavior via a finite state machine ("an automaton") that models the allowed operation sequences. Invocation of the service must satisfy these sequences. For example, a seller may require a buyer to log in before ordering something. Thus, we can describe valid sequences of operation invocations. The interactions between the service operations occur through message exchanges. These messages are the inputs and the outputs of the operations. We can describe the automaton representing the service behavior as: a black box which conveys a relation between its inputs and its outputs (external behavior) and as a white box which indicates the required interactions between the service operations to provide outputs from given inputs (internal behavior).

The automaton describing the service behavior is composed of a set of states and transitions between the states. Each service operation is defined by a state. For a current state, the successor state belongs to the set of its matching states: i.e., the output of the current state can be connected to the input of its next. Therefore, a transition connects a state to one of its valid next state. An interaction between operations can be performed only if it is defined in the automaton.

5 Service composition

The composition of services is critical since it provides novel services and novel functionalities to the client. By combining services, the client (a learner/teacher or a researcher) can obtain a desired output from a given input not provided by any service but by several combined services. In our approach, we consider that two services can be composed if an ouput of one is equal to an input of the other one. Figure 1 shows an example of two services that are composed in figure 2. We describe each service by an automaton. The service "tex_transf" has an operation "latex" that transforms a "tex" file into a "dvi" file. This operation has one input ("tex") and two outputs ("dvi" and "error"). The service "dvi_transf" has the operation "dvips" that transforms a "dvi" file into a "ps" file. Therefore, This operation has an input "dvi" and an output "ps". These two services can be composed because they have a same input/output ("dvi"). Figure 2 describes the composed service that provides novel functionality. As a black box this composed service one input ("tex") and outputs ("error", "dvi", "ps"). As a white box, it has two operations : "latex" and "dvips". The interactions between them are presented in Figure 2.

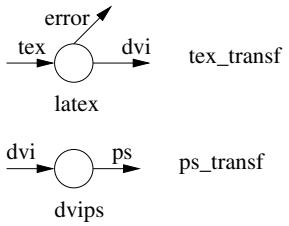

Figure 1. Example of two services

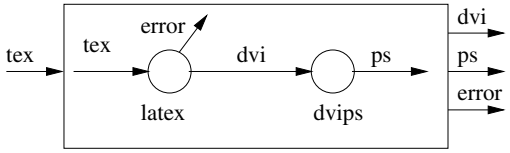

Figure 2. The composed service

We have proposed three models of service composition [6]: static that allows to combine service offers defined at compile-time, semi-dynamic that allows to combine service types (a service offer is an instance of a service type) defined at compile-time and dynamic which allows to combine services defined at run-time.

6 Application

We present an application we have developed using our approach to discover and describe documents. It allows documents to be reused and shared by using the approaches defined for web services. A client can discover a document which is downloadable on the client machine with no interaction with its provider. A document doesn't have an interface. This approach is used in the semantic web domain. Our idea is to direct and view a document as a service, i.e., a document accessible via through an interface. This allows

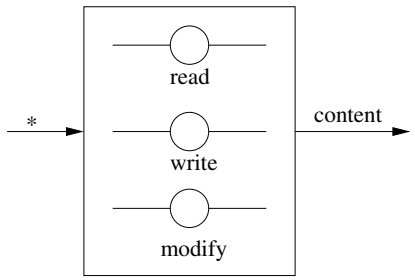

Figure 3. An automaton describing the document behavior

clients to use the same type of interface to interact with different documents (web documents, databases, etc.). In this context, we can add an interface to the document that allows clients to read, write and modify a document.

We describe a document with our proposed metadata (Section 2) as :

- A learning resource: title, version, etc. We can use the metadata proposed by Dublin Core [1] to describe documents.
- A service helping researchers : it includes the description of the research area of the document, the description of its author, and so on.
- A service in general: it includes delivery and request channels, payment and pricing and so on.
- By behavior: we describe the inputs/outputs of a document. We consider that a document doesn't have an input. A client doesn't need inputs to interact with a document. The output of a document can be its content. Figure 3 describes the document behavior with an automaton. The three operations (read, write, modify) don't have inputs ("*") and can provide as outputs the content of the document after reading, writing or modifying.

7 Related Work

Many different approaches for describing and discovering services have been developed. For example, DAML-S [10] provides a set of characteristics that allows describing a service. This description addresses only the static properties of a service. It does not describe the behavior of a service and its interface. In our approach of service indexing, we have used this description as a reference to characterize a service by static properties. Our approach is more powerful than DAML-S since it provides complementary properties describing the service by its behavior and its interface. These properties can be used by clients to query and find a service in distributed systems.

To discover a service, we can select for example UDDI which, is a registry that allows a Web service to be discovered via a yellow page style of search. It allows a service to be discovered by querying only its static properties. It provides a static schema for service description and the service provider cannot modify this schema or create databases to advertise its services offers. Then, in our approach, each service provider can define its databases and can advertise its service offers by using their static properties, their behaviors and their interfaces. The clients use the same interface to query these three levels of service description. So, the service indexing and discovery become more sophisticated.

Many different approaches for service composition have been developed like [14], [15], [16] but it is not clear that they compose services based on their behaviors. Our approach of semi-dynamic [6] composition model is the first one that exceeds the performance and complexity problem of the dynamic composition model.

8 Perspectives and Conclusions

In this paper, we have presented our current related work in the indexing and searching of learning and research services. For this latter operation, we have proposed a metadata model for describing learning services. The current model that we have proposed, describes a learning service with three dimensions: as a learning resource, as a service that contribute and help researchers and as a general service. This model is a powerful one since it provides a complete description of learning services and extends services to be used by researchers/teachers and not only by teachers. It is flexible because it ensures compatibility with existing approaches like IEEE LOM, Dublin Core and DAML-S. The clients and the servers of this metadata model can be teachers, researchers and students. We have implemented this model with ontologies, consisting of concepts representing the characteristics of a service. Flexibility is introduced since the ontology and its content can be changed and adapted to usage.

We have presented a trader we designed and developed: it is based on ontologies and knowledge representation. It allows a service provider to advertise a learning service offer and a client to discover a learning service by querying its three dimensions of description. By using ontologies and a logic language, the service discovery is more flexible. This requires that the ontology be designed with care and user intention in mind.

Based on SDL and Interface Automata, we can describe the service behavior which enables the client to use the service discovered and to understand how the invocation of the service operations needs to be performed. Based on the service behavior description and the trader, we have presented in this paper an approach that allows the com-

position of services in distributed systems. This approach is a powerful one since it allows the client to benefit from the functionalities of more than one service to get novel functionalities and novel services.

As perspectives, we will propose an approach of trader federation to compose services. In this context, when a client queries a service from a trader and this service doesn't exist, the trader can interact with other traders to search and compose services to satisfy the client's request. So, this approach will allow the composition of services located on different traders.

References

[1] A. Powell. Dublin Core in RDF. Technical Report, 1998.

[2] IEEE. Draft standard for learning object metadata. In *http://ltsc.ieee.org/wg12/*. Technical Report, 2002.

[3] *www.uddi.org*.

[4] O. K. Zein and Y. Kermarrec. An Elaborate and Flexible Trader Based on Ontologies. In *IFIP WG6.7 Woskshop and EUNICE Summer School on Adaptable Networks and Teleservices*, pages 103–108, Trondheim, Norway, September 2002.

[5] T. R. Gruber. A Translation Approach to Portable Ontology Specifications. *Knowledege Acquisition, 5 : 199-220*, 1993.

[6] O. K. Zein and Yvon Kermarrec. Static, semi-dynamic and dynamic composition of services in distributed systems. In *IEEE International Conference on Internet and Web Applications and Services (ICIW'06)*, Guadeloupe (French Caribbean), 17-25 February 2006.

[7] M. Dumas, J. O'Sullivan, M. Heravizadeh, D. Edmond, and A. Hofstede. Towards a semantic framework for service description. *In Proceedings of the 9th International Conference on Database Semantics, Hong-Kong*, April 2001.

[8] M. Merz, M. Witthaut, and S. McConnel. Catalogue and Service Architecture. http://osm-www.informatik.uni-hamburg.de/osm-www/public/docs. *OSM D8*, 1997.

[9] W. Ng, G. Yan, and E. Lim. Heterogeneous product description in electronic commerce. *ACM SIGeCom Exchanges, 1(1):7-13*, 2000.

[10] *http://www.daml.org/services/*.

[11] D. Fensel, S. Decker, M. Erdmann, and R. Stude. Ontobroker : The very high idea. *In Proceedings of the 11th International Flairs Conference (FLAIRS-98), Sanibal Island, Florida, USA*, May 1998.

[12] ITU-T. ITU-T Recommendation Z.100. Specification and Description Language (SDL). 1989.

[13] L. De Alfaro and T. A. Henzinger. Interface Automata. In *Proceedings of the Ninth Annual Symposium on Foundations of Software Engineering*. ACM Press, 2001.

[14] J. Yang, M. P. Papazoglou, and W. den Heuvel. Tackling the Challenges of Service Composition in E-Marketplaces. In *The 12th International Workshop on Research Issues in Data Engineering : Engineering e-Commerce/e-Business Systems*. RIDE 2002.

[15] M. Pistore et al. Planning and monitoring web service composition. In *AIMS*, 2004.

[16] D. Berardi et al. Automatic service composition based on behavioral description. In *IJCIS*, 2005.