CONTEXT AWARE SERVICE ORIENTED ARCHITECTURE FOR WEB BASED EDUCATION

Stephen J.H. Yang  
Blue C.W. Lan  
Irene Ya-Ling Chen  
Ben J.D. Wu  
April C.N. Chang  
National Central University, Taiwan  
National Central University, Taiwan  
Ching Yun University, Taiwan  
National Central University, Taiwan  
National Central University, Taiwan  
jhyang@Lst.ncu.edu.tw  
blue@Lst.ncu.edu.tw  
irene@cyu.edu.tw  
wallace@Lst.ncu.edu.tw  
apriller@Lst.ncu.edu.tw

ABSTRACT
In the past years, most of e-learning researches devoted themselves to the exchange and interoperability of various educational materials such as LOM and SCORM. Recently, IMS Learning Design specification provides a generic and flexible language to support pedagogical diversities in online learning. In this paper, we propose a context-aware SOA to enable the reusability of both learning designs and learning materials by treating them as Web Services. We utilize OWL-S as the service description language to support reasoning procedures and teachers can utilize the semantic matchmaker to retrieve desired learning designs and learning materials in a semantic manner so that they can compose their learning scenarios more efficiently and flexibly.

KEY WORDS
IMS Learning Design, Context-aware, SOA, OWL-S

1. Introduction
With the evolution of computer technologies, lots of learning systems are invented for different pedagogical purposes. These e-learning systems bring a great of benefits such as asynchronous interactions, group collaborations, individualized instructions, distance learning etc. and all learning materials can be accessed anytime, anywhere in a cost-effective manner. In the past years, many researches contributed to promote exchange and interoperability of educational materials among different e-learning systems including IEEE LTSC’s Learning Object Metadata (LOM) [1], ADL’s Sharable Content Object Reference Model (SCORM) [2] etc. Recently, IMS Global Learning Consortium released the Learning Design specification [3] to support the use of a wide range of pedagogies while enabling the reusability of learning materials in online learning. However, the more power the specification provides, the more complexity the description is and IMS Learning Design is no exception. It is difficult for teachers to compose complete learning scenarios from scratch when they want to perform some pedagogical approaches on the Web. Similarly, it will be difficult for teachers to prepare all learning materials that may be presented in different modalities. It becomes a major challenge for e-learning society to share and reuse all sorts of learning designs and learning materials.

The emerged Web Services oriented technologies provide a flexible, efficiency and loosely coupled way to reuse existing applications and researches. Web Services technology concentrates not only on interoperability but also on how to describe, publish, locate and invoke Web Services. A number of standards and specifications created from industry and academia have contributed to the development of Web Services such as WSDL[4], UDDI[5]. Service providers can describe services with WSDL to specify what the service does and how to accomplish the service. UDDI creates a standard interoperable platform that enables companies and applications to quickly, easily, and dynamically locate Web Services over Internet. However, the above building blocks of Web Service infrastructure can only behave well in syntactical level. The Web Service infrastructure lacks of semantic information to support inferential capability.

Information on current Web is designed primarily for human rather than for machines. Semantic Web needs some other languages to annotate Web contents with unambiguous descriptions besides XML. Languages such as RDF [6], DAML+OIL [7] and recently OWL [8] provide predicate like or description logic based annotation of Web contents to encourage inferential procedures on annotated Web contents. The combination of Semantic Web’s expressive power and Web service infrastructure results in the concept of Semantic Web Services and the vision of Semantic Web Services are to facilitate the automation of Web Service tasks such as automated Web Service discovery, composition, interoperation and execution monitoring. OWL-S [9] is the first step toward Semantic Web Services and it supplies Web Services providers with a core set of markup language constructs for describing the properties and capabilities of their Web Services in an unambiguous, machine interpretable way.

In this paper, we propose a context-aware service oriented architecture (SOA) for Web based education to help teachers reuse existing learning designs and learning materials in a semantic manner. There are three building blocks in the proposed architecture as shown in Figure 1: Agent Platform, Service Pool and Semantic Matchmaker. Teachers and services providers can utilize Request Agent and Service Agent to perform tasks on behalf of them and Broker Agent will be responsible for the coordination of
all tasks. In order to enable capability matching between requests and services, the Service Pool is designed to encompass a general UDDI Registry and Semantic Profiles that store semantic descriptions of services. The Semantic Matchmaker will employ the Reasoner and Domain Ontologies to perform inferential procedures and it will activate the Planner to decompose user’s request if there is no services fulfill the request.

We will use educational materials and learning materials interchangeably and simplify the example codes for readability in the paper. The remainders of the paper are organized as follows: Section 2 addresses related works. Section 3 presents our context-aware service oriented architecture for Web based education. We conclude this paper and address future research in Section 4.

2. Related Works

Rather than capturing the specifics of many pedagogies that may require many specialized implementation of both design and runtime systems, IMS Learning Design extends EML [10] developed at OUNL to provide a generic and flexible language by balancing the generality and pedagogic expressiveness. IMS Learning Design provides the capability of designing any unit of learning that may involve several roles such as students, tutors and examiners etc., and these units of learning can be coordinated as learning flows to realize expected learning objectives. IMS Learning Design not only makes a clear distinction between pedagogical approaches and educational materials but also separates elements of unit of learning from each other such as activity, activity structure, roles and resources etc. This loosely coupled design makes a lot of elements more reusable than elements in existing Web based education systems.

OWL [11] is a formal language designed to describe Web contents in an unambiguous, machine interpretable way. OWL-S is a specific application for Web Services by applying OWL on service’s capability representation. OWL-S provides declarative publications of services properties and capabilities, API for Web Services, specifications of prerequisites and consequences of individual services and descriptors for the state of services execution for automatic services discovery, invocation and execution monitoring. Agents can perform some tasks such as decision making on behalf of people based on the expressive power of such formal language. For the sake of realizing the above goals with agent technologies, researchers from CMU have developed the DAML-S/UDDI Matchmaker that expands on UDDI to provide semantic capability matching [12]. The matchmaker can perform inferences on the subsumption hierarchy leading to the recognition of semantic matches regardless of their syntactical differences. They adopt a flexible matching strategy based on inputs and outputs to identify similarities between requests and service’s advertisements. There are four different matching degrees:

1. Exact match: Both outputs and inputs of desired service are equivalent of request’s, i.e. IN_{Ad} = IN_{Req} and OUT_{Ad} = OUT_{Req}.
2. Plug- In match: It indicates that outputs of advertisement are more specific than request’s or inputs of advertisement need less information than request provides, i.e. OUT_{Ad} \supset OUT_{Req} or IN_{Req} \supset IN_{Ad}.
3. Subsumed match: It denotes that outputs of advertisement can only provide partial information needed by request or inputs of advertisement are more specific than request provides, i.e. OUT_{Req} \supset OUT_{Ad} or IN_{Ad} \supset IN_{Req}.

Although DAML-S/UDDI Matchmaker expands functionality of UDDI registry to enable capability matches, they did not take context into account. Requesters may still receive lots of invalid services and they need to verify all matched services against their contexts manually.

LARKS [13] matchmaking processes both syntactic and semantic matching and they identified different degrees of partial matching as well as CMU’s DAML-S/UDDI Matchmaker by combining different filters. They defined five filters including Context Matching, Profile Comparison, Similarity Matching, Signature Matching, and Constraint Matching. Requesters may select any desired combination of these filters according to different concerns such as efficiency, computation cost etc. In order to get more precise services for requests, both of previous approaches need to spend considerable time on matching between requests and services so that requesters may have a trade-off between the efficiency and the precision by using their approaches.

3. Context Aware SOA for WBE

In order to encourage reusability of IMS Learning Design and various electronic educational materials so that teachers can create different pedagogies along with desired educational materials, we propose a context aware SOA for Web based education as illustrated in Figure 1. There are three building blocks in the proposed architecture: Agent Platform, Service Pool and Semantic Matchmaker and teachers can obtain desired services by going through service discovery and service invocation as specified in the following steps:

1. Teachers submit related information to Request Agent.
2. Request Agent sends the request to Broker Agent.
3. Broker Agent dispatches the request to Planner.
4. Planner activates Reasoner to determine whether some services fulfill the request.
5. If the request is unmatched, Planner will access Domain Ontologies to decompose the request into smaller sub-requests and perform step 4 for each sub-requests.
6. Planner returns matched services to Broker Agent and Broker Agent will relay the information to Request Agent.
7. Teachers submit required inputs for desired service to
Request Agent.
(8) Request Agent sends required inputs to Broker Agent and Broker Agent will relay the information to Service Agent.
(9) Service Agent will construct a binding between teachers and service providers.

We will specify each component of the proposed architecture more detail in the following sub-sections to demonstrate how we perform the above steps for services discovery and invocation.

Figure 1. Context Aware SOA for WBE

3.1 Agent Platform

The Broker Agent will coordinate the matching process and service discovery and invocation so that teachers do not need to communicate with service providers. The proposed Agent Platform can reduce communication efforts for service requesters and service providers through the Request Agents, Broker Agents, and Service Agent.

Request Agent will perform service requesting and service invocation on behalf of teachers by collecting related information from teachers and formatting collected data as the user request specification. These information may include teacher’s identity, required service name, required service category and required service quality etc. Figure 2 shows an example of user request.

<icu:ServiceRequest>
  <icu:Requester>
    <dc:identifier>blue@Lst.ncu.edu.tw</dc:identifier>
    <icu:RequesterName>Blue</icu:Name>
  </icu:Requester>
  <icu:RequiredService>
    <icu:ServiceName>PBL</icu:ServiceName>
    <icu:ServiceCategory>
      <icu:CategoryName>NAICS</icu:CategoryName>
      <icu:CategoryValue>611710</icu:CategoryValue>
    </icu:ServiceCategory>
    <icu:ServiceContext>
      <icu:Availability>High</icu:Availability>
    </icu:ServiceContext>
    <icu:ServiceIO>
      <icu:Output>
        <icu:Parameter>IMS Learning Design</icu:Parameter>
      </icu:Output>
    </icu:ServiceIO>
  </icu:RequiredService>
</icu:ServiceRequest>

Figure 2. Specification of user request in XML

Broker Agent is responsible for the coordination among Request Agent, Service Agent and Semantic Matchmaker. It needs to check the identities of service requesters and service providers to determine whether their requests are permitted against their authorizations. In addition, Broker Agent also needs to perform some balance policy for dispatching service requests to different Semantic Matchmakers.

Service Agent will receive the invoking request sent from Request Agent through Broker Agent and check the invoking request to ensure required information are included. Then, Service Agent will retrieve required inputs from the invoking request and help the binding between two parties.

3.2 Context Aware Service Pool

We apply OWL-S as the service descriptions language to support semantic matching processes in the proposed architecture so we need an extra repository called Semantic Profile for the storage of semantic descriptions except for a UDDI Registry. For service publishing, the service description will be divided into two parts: grounding and others, and the grounding will be registered into the UDDI Registry first. Then, other parts will be registered into the Semantic Profiles with ID key derived from former UDDI registration. For service description, we also take service’s context into account and design two contextual profiles: Device and QoWS Profile, to empower context-aware matching process.
Different services may offer various modalities of learning materials and services providers can indicate required runtime environment for provided services. For example, the provider may offer video services and he can specify that requester’s runtime environment should support provided media type in order to deliver the service successfully. Thus, we extend OWL-S to specify developed contextual profiles as illustrated in Table 1 and register them into the Semantic Profile. Figure 3 is the corresponding specification of contextual profiles in OWL-S.

<table>
<thead>
<tr>
<th>Table 1. Schema of Contextual profiles</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Device Profile</strong></td>
</tr>
<tr>
<td>1) RequiredDisplayResolution:</td>
</tr>
<tr>
<td>Service providers can specify minimum</td>
</tr>
<tr>
<td>resolution needed to present the</td>
</tr>
<tr>
<td>outputs of the service.</td>
</tr>
<tr>
<td>2) RequiredMediaType</td>
</tr>
<tr>
<td>Service providers can specify what</td>
</tr>
<tr>
<td>kind of media types will be returned.</td>
</tr>
<tr>
<td><strong>QoWS Profile</strong></td>
</tr>
<tr>
<td>1) MaxProcessTime</td>
</tr>
<tr>
<td>The maximum processing duration taken</td>
</tr>
<tr>
<td>by the service.</td>
</tr>
<tr>
<td>2) RequiredBandwidth</td>
</tr>
<tr>
<td>Service providers should setup a</td>
</tr>
<tr>
<td>threshold to evaluate the minimum</td>
</tr>
<tr>
<td>bandwidth in order to complete the</td>
</tr>
<tr>
<td>service in a reasonable time.</td>
</tr>
<tr>
<td>3) Reliability</td>
</tr>
<tr>
<td>Service providers should evaluate the</td>
</tr>
<tr>
<td>ratio of successful execution and we</td>
</tr>
<tr>
<td>define it as the ratio of successful</td>
</tr>
<tr>
<td>execution in one hundred invocations.</td>
</tr>
<tr>
<td>4) Availability</td>
</tr>
<tr>
<td>Service providers should evaluate the</td>
</tr>
<tr>
<td>amount of available time in an interval</td>
</tr>
<tr>
<td>and we define it as amount of available</td>
</tr>
<tr>
<td>minutes in an hour.</td>
</tr>
<tr>
<td>5) Cost</td>
</tr>
<tr>
<td>The price of service execution.</td>
</tr>
</tbody>
</table>

Figure 3. Contextual profile in OWL-S

3.3 Ontology Based Semantic Matchmaker

We utilize ontology technology to support inferential procedures on user request and employ a Planner to decompose user request into smaller sub-requests based on Domain Ontologies if the original request are not matched. The decomposition process is based on the subsumption relationships specified in related ontologies. For example, if a teacher wants to compose a pedagogical approach called problem based learning (PBL) by reusing an existing IMS Learning Design and he may submit a request as shown in Figure 2. If the request cannot be fulfilled, Planner will infer Domain Ontologies to generate a PBL hierarchy by parsing subsumption and equivalence relationships such as subClassOf, equivalentClass and sameAs descriptions in OWL-S based ontologies. Figure 4 illustrates an example of PBL hierarchy.
According to Figure 4, the original request may be divided into three sub-requests: brainstorm, discussion and grouping and Reasoner will perform capability match for each sub-request. In order to improve matching performance, we develop a three-stage matching process between the request and services. The first stage matches the request and services upon specified service category and contextual requirements. Teachers may utilize some famous taxonomy such as NAICS [14] and UNSPSC [15] to get the service category of desired services and specify them in the request. Reasoner will perform keyword search of the service category to quickly retrieve expected services. The second stage will perform input and output match (IO match) between the request and services retrieved in the first stage. We adopt CMU’s definition of four matching degrees and only plug-in and exact matches will be accepted. The third stage will match the request with services qualified from second stage upon service name. Similarly, only plug-in and exact matches are accepted. For example, if we have two IMS Learning Design services that named “problem based learning” and “Brainstorm” respectively as illustrated in Figure 5, a teacher may submit a request like Figure 2 to obtain an IMS Learning Design of PBL. In the first stage, the two services will be retrieved from Service Pool because their service categories and contextual requirements are identical to the request. In the second stage, two services are also qualified based on the exact match of the output. However, only “problem based learning” service will be returned to the teacher because “Brainstorm” service gets a subsumed match based on the PBL ontology as illustrated in Figure 4 in the third stage.

4. Conclusion and Future Research

We proposed a context aware SOA for WBE to help teachers compose expected pedagogical approach along with desired learning materials by facilitating automatic discovery of IMS Learning Design and educational materials. Semantic Profile is invented to accommodate semantic descriptions and it is utilized as the auxiliary of UDDI Registry to support inferential procedures. In addition, we also develop a three-stage matching process to improve the matching performance and we will
conduct the evaluation to prove the enhancement of the performance in the future.

5. Acknowledgement

This work is supported by Ministry of Education, Taiwan under grant A-91-E-FA06-4-4.

References: