Distributed E-learning Based on SOA

Khosro Bahrami, Behzad Daemi and Mehrnoush Abedi
Niroo Research Institute
Iran

1. Introduction

In this chapter, we propose distributed architecture for e-learning presentation and generation based on Service Oriented Architecture (SOA). In first, we present Web based architecture for the distributed e-learning system on SOA. In this architecture, e-learning systems are composed of services such as LMS, Learning Object Service, Learning Content Management Service (LCMS) and so on those are published through Learning Service Providers on the Web. E-learning systems can publish or use these services to share their applications and contents. In our proposed distributed e-learning architecture, we show a complete solution for distributed e-leaning based on multi-layered architecture that each layer has specific purpose to aggregate and present services. In this architecture, service providers, service requesters and service brokers are described. In Following, we will represent some sections about how to generating a course for the e-learning system. According to our experience in using web based e-learning systems, course generation is a time consuming task meanwhile the course is the most important part of the system.
An e-learning system could not be successful without representing high quality courses. Because technology is changing rapidly, time is an important factor in creation a course, especially in areas where knowledge and skills change rapidly, such as in the IT domain, when the course is finally finished and ready to publish, the content of the course is out of date. Therefore, using automatic course generation beside the conventional methods of course authoring could be an inevitable solution. The goal of section 7 (Distributed course generation) section is representing a way to better usage of myriad electronic documents those are freely available in making tutorials and helping researchers in extracting and refining their required information from data centers and World Wide Web.
In some sections of this chapter, we are going to represent the usage of semantic Web, Knowledge Objects and Learning Objects in generating, maintaining and updating the courses.

2. Service Oriented Architecture and Web Service

Service Oriented Architecture (SOA) is an architecture that functions are defined as services and are accessible through their interfaces. This architecture enables rapid creation of

distributed application from distributed services base on their functionality. SOA consists of services. The services of SOA are software entities with discoverable interface to provide certain functionality.
SOA consists of three entities, Service Provider, Service Broker and Service Requester (Figure1).

- Service Provider creates the services and publishes them by registering the implemented services in Service Broker.
- The services are registered in Service Broker based on contract.
- The Service Requester finds the service in Service Broker and binds their applications to the Service Provider.

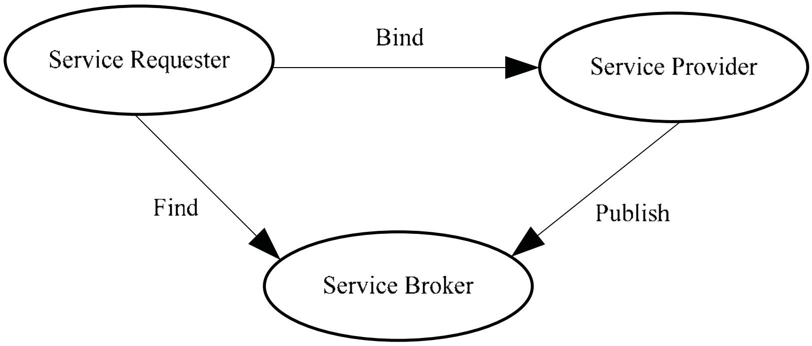

Fig.1. Service Oriented Architecture

Web Services are systematic and extensible framework for applications interaction and comprise of SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language) and UDDI (Universal Description Discovery Integration) that can be accessed via Internet. Web Services architecture is independent of the platform and language, so it is usable among various applications and frameworks. Implementation of Web Services means data and operations structuring inside an XML document that complies with the SOAP specification. SOAP is a protocol specification that defines method of exchange XML encoded data between platforms. It also defines a way to do remote procedure calls (RPCs) using HTTP as the foundation of communication protocol. Web Service is used as a technology to implement SOA (Curbera et al., 2002), (Huhns et al., 2002), (Zheng et al., 2008). Entities of SOA can be implemented through Web Services. For this purpose, Service Provider describes the Web Services function with WSDL document. The description includes data type definitions, operations, input/output, network addresses and so on. Service Broker used UDDI specification to register and locate web services in Service Provider. UDDI is a web service based on XML and SOAP.

3. Architecture of Distributed E-learning based on SOA

As described in the previous section, the SOA is composed of three entities. So, the architecture of distributed e-learning based on SOA has three entities as described below (Figure2) (Bahrami et al. 2007):

3.1 E-Learning Portal

E-Learning Portal acts as service requester of SOA. This center gets the services of e-learning from E-learning Service Centers on the web and interacts with e-learning users to provide e-learning services. Communication of E-learning Portal and E-Learning Service Center is done on the web through SOAP and XML protocol.

3.2 E-Learning Service Center

E-Learning Service Center acts as service provider of SOA. This center provides the services of e-learning for E-Learning Portals. E-Learning Portals search the services on the web to find the required services. So, the E-learning Service Centers should be registers their services in Registration Centers.

3.3 E-Learning Registration Center

E-learning Registration Center acts as service broker of SOA. Services of E-learning Service Centers are registered in this center. This center provides WSDL document to publish and search service description for the UDDI. E-learning Portal finds the service in E-learning Registration Center and binds the service from E-Learning Service Center.

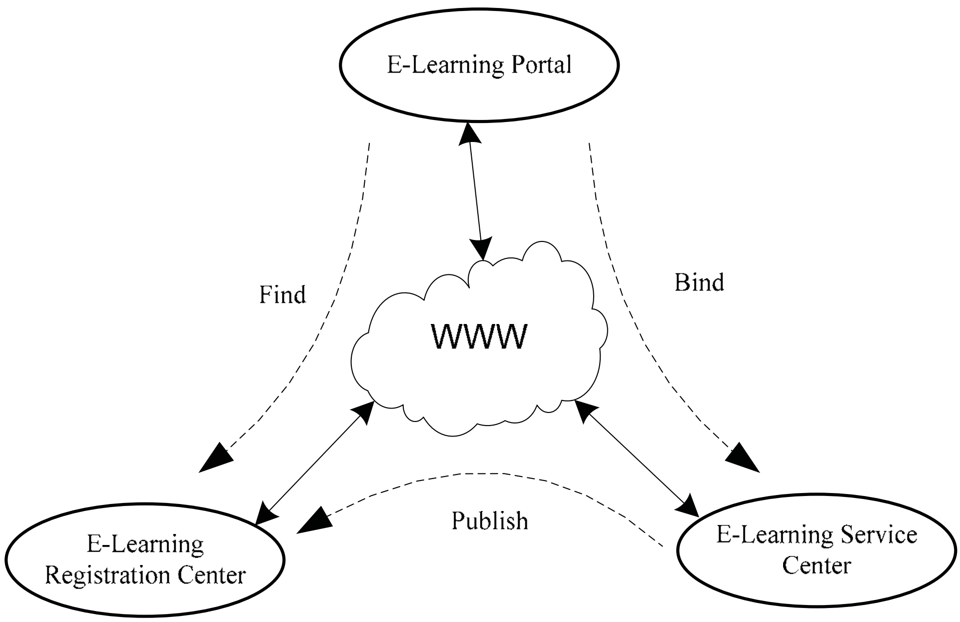

Fig.2. Distributed E-learning based on SOA

4. E-learning Portal as Service Requester

In this section, we propose multi-layer architecture of e-learning portal. The proposed e-learning portal has 3-tier architecture (Bahrami et al. 2007), includes Front-End tier, Middle-Ware tier and Back-End tier (Figure3):

4.1 Front-End Tier

Front-End tier provides customized and personalized web based user interface of the e-learning system. The contents of presented web pages are aggregate from various sources. This tier provides interface of e-learning services. E-learning services have implemented in Middle-Ware tier and include six services, namely, Course Content Design Service, Course Content Offering Service, Course Management Service, Learning Registration Service, User Management Service and Collaboration Service.

Front-End tier communicates with Middle-Ware tier in a web-based environment through web services. For this purpose, this tier initiates SOAP messages and requests information from services of Middle-Ware tier. In the reverse direction, the Front-End tier collects and integrates the SOAP response results obtained from Middle-Ware services and represents to users (Cheng et al., 2006), (Fu et al., 2005).

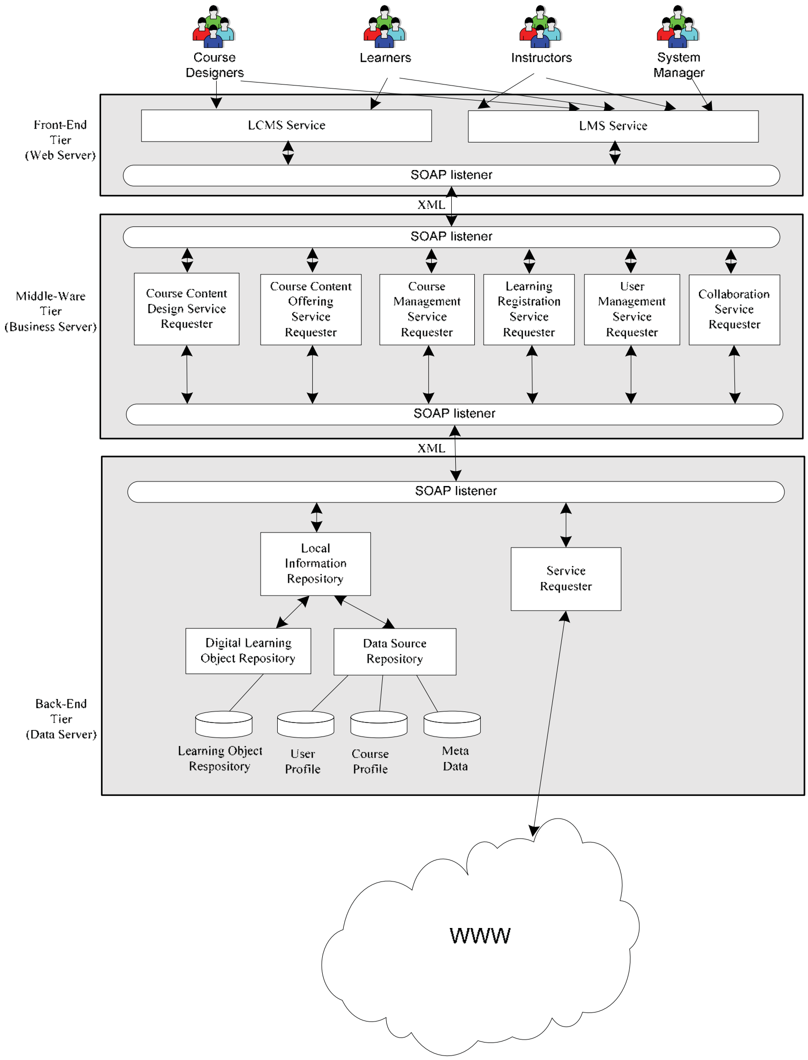

Fig.3. Architecture of E-learning Portal

4.2 Middle-Ware Tier

Middle-Ware tier is the engine of the portal e-learning system and provides logical and control functions of the e-learning system for Front-End tier. For this purpose, this tier searches and provides services of e-learning from e-learning service providers on the web. This tier supports six services, namely, Course Content Design Service, Course Content Offering Service, Course Management Service, Learning Registration Service, User Management Service and Collaboration Service. So, this tier uses a requester module for each service to get the information from services on the web and provide the functionality for Front-End tier. These requester modules are described below:

- Course Content Design Service Requester

This service requester gets the functionality of Course Content Design Service in service providers to provide designing capabilities for content authors and course designers (Vossen et al., 2006).

- Course Content Offering Service Requester

This service requester gets the functionality of Course Content Design Service in service providers to provide the learning contents to learners. Learning contents are composed of learning objects that subsequently are forwarded to the learners and presented at client browser.

- Course Management Service Requester

This service requester gets the functionality of Course Content Design Service in service providers to define new course, remove a course and update course information.

- Learning Registration Service Requester

This service requester gets the functionality of Course Content Design Service in service providers to register the learner information and course information in course registration database.

- User Management Service Requester

This service requester gets the functionality of Course Content Design Service in service providers to define new user, remove a user and update user information. It also provides the user information to the Learning Registration Service.

- Collaboration Service Requester

This service requester gets the functionality of Course Content Design Service in service providers to provide the collaboration tools and environments (e.g. messages, forum, email) to the users to provide loosely connection for users.

In addition, Middle-Ware tier includes Learning Object Aggregation Engine. Learning Object Aggregation Engine provides a global view of distributed learning contents and learning objects on learning content providers. The global view is obtained from the global Metadata schema. This schema shows virtual structure and navigation of course content and learning objects. It also provides the physical storage site of learning objects on learning object repositories. This service aggregates the learning content and learning objects for Course Content Design Service, Course Content Offering Service and Course Management Service (Guiling et al., 2005).

4.3 Back-End Tier

This tier includes Local Information Repository and Service Requester to provide learning content and learning objects to upper tiers. These centers offer digital learning objects and learning contents and related local schema. The schema includes the global information about learning objects and digital learning contents. As mentioned above, web services are used for implementation of learning object repositories to deliver digital learning content to services of e-learning systems (Navaro et al.,2006), (Liao et al., 2004). This tier has composed of Information Repository and Service Requester.

- Local Information Repository

Local Information Repository provides the digital learning object repository and data source (user profile and course profile) to the services and components of the LMS/LCMS subsystems. This subsystem delivers information to LMS/LCMS applications of e-learning systems using web services.

Digital learning object repository includes Meta data repository and learning objects. Meta data repository includes the global information about learning object repositories and digital learning objects and global schema. Meta Data Repository provides a global view of distributed learning contents and learning objects. This schema views virtual structure and navigation of course content and learning objects, but doesn't show the physical storage site of learning objects on learning object repositories. This service aggregates the learning content and learning objects for Course Content Design Service, Course Content Offering Service and Course Management Service of Middle-Wire tier.

Data source repository includes heterogeneous data sources that are not designed with the focus on sharing, such as course profile, user profile and so forth.

- Service Requester

Service Requester includes web sites and other e-learning systems that provide the e-learning services to components of the LMS/LCMS subsystems. As mentioned above, SCORM standard and Web Services are perfectly achievable for implementing the learning object repositories and delivering the digital learning content to LMS/LCMS applications of e-learning systems. These repositories are any learning object repositories on the Web that support SCORM standard (Li et al., 2005), (Madhour et al., 2006), (Frosch et al., 2004), (Shih et al., 2006).

5. E-Learning Service Center as Service Provider

E-Learning Service Center includes web sites and other e-learning systems that provide the various kinds of e-learning services such as Course Content Design Service, Course Content Offering Service, Course Management Service, Learning Registration Service, User Management Service and Collaboration Service to components of the LMS/LCMS subsystems. These services are described below:

- Course Content Design Service

This service provides designing capabilities for content authors and course designers. These capabilities include course structure design, course content and learning object design with SCORM standard (Vossen et al., 2006).

- Course Content Offering Service

This service is responsible to provide the learning contents to learners. Learning contents are composed of learning objects that subsequently are forwarded to the learners and presented at client browser. It also tracks the learner activities and updates the learning status in learner profile during the learning process. Another capability of this service is assessment of learners. This service uses the content of Learning Object Repository to provide learning objects.

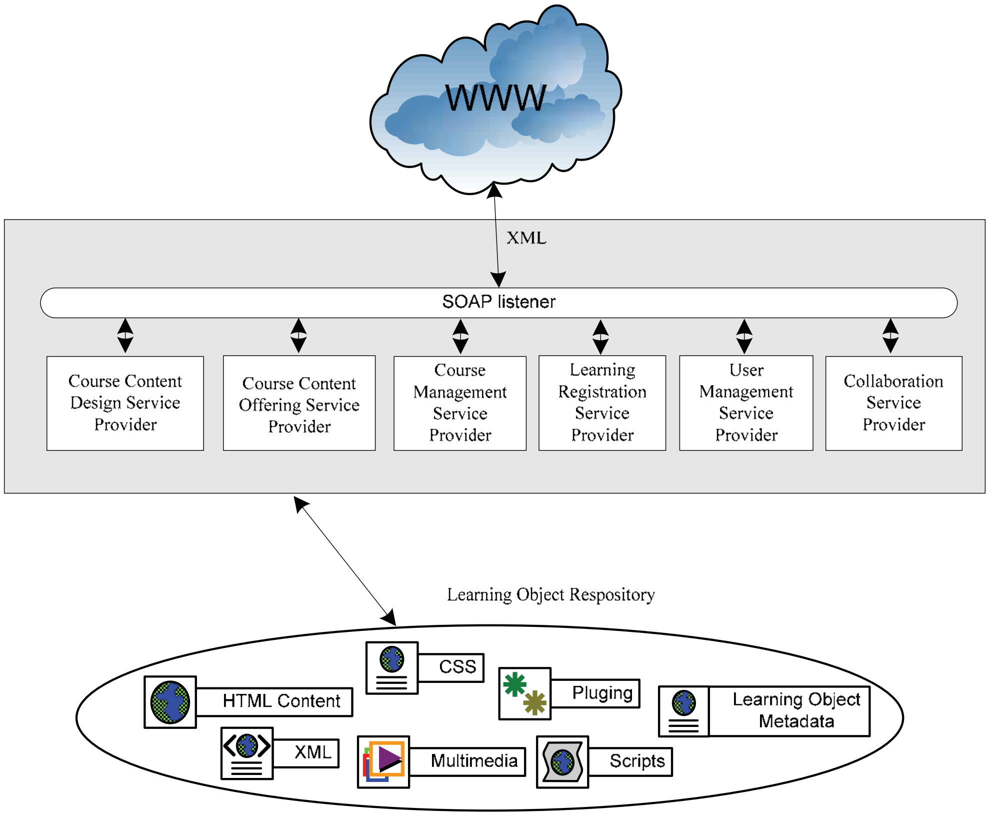

Fig. 4. Architecture of E-Learning Service Center

- Course Management Service

This service provides the capabilities to define new course, remove a course and update course information. It also provides the course information to the Learning Registration Service.

- Learning Registration Service

Once a learner selects a course for learning, the Learning Registration Service registers the learner information and course information in course registration database.

- User Management Service

This service provides the capabilities to define new user, remove a user and update user information. It also provides the user information to the Learning Registration Service.

- Collaboration Service

This service provides the collaboration tools and environments (e.g. messages, forum, email) to the users to provide loosely connection for users.

6. Why distribution?

In regard to the large volumes of data, redundancies and processing power, distribution of Learning Object Repositories is needed in an e-learning system. In the content provider architecture (figure 5), non stop knowledge object generating and updating tasks are running in each of the lowest level nodes. Mediators are also busy with continuous knowledge object enrichment and Learning Objects generation. Without using distributed architecture all of the above heavy I/O and CPU band tasks ought to run in the content provider node that is not economical.

7. Distributed Course Generation

Producing courses on a distributed network is depended on the employed architecture. Our proposed architecture is depicted in figure 5. According to this architecture, we have chosen a tree like distribution over the chosen medium (e.g. Power line networks, wireless networks or other employed media in World Wide Web). At the lowest level, there are some Learning Object providers and at the middle level there is

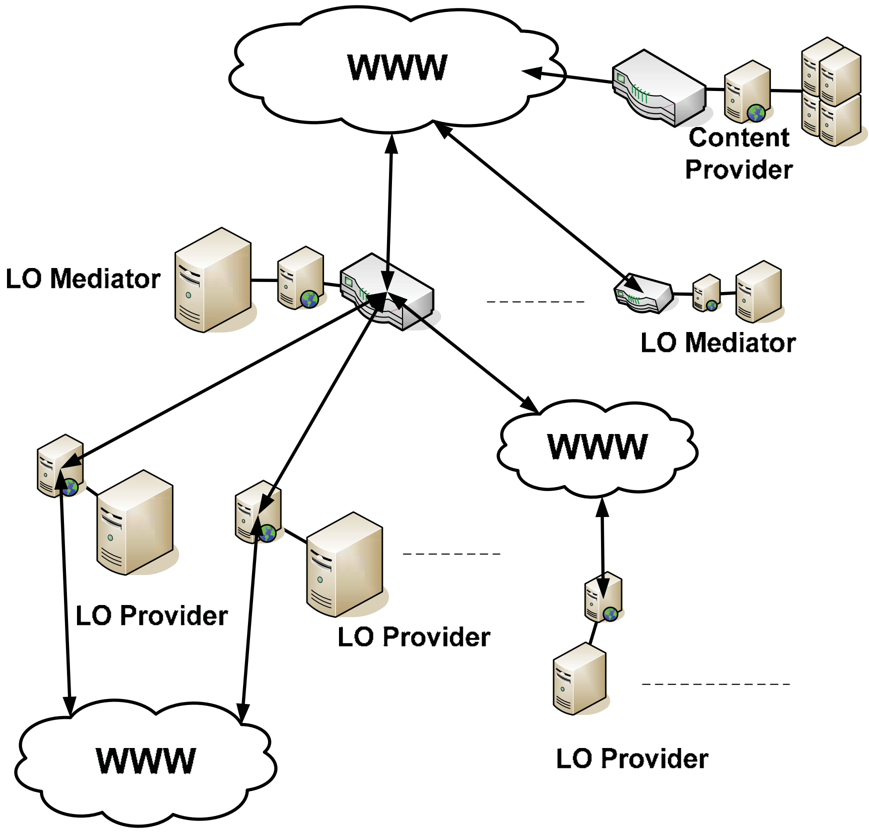

Fig.5. Architecture of the distributed content provider

a Learning object mediator for providing Learning Objects for highest level that is the content provider node. Services those are running at this node provide courses contents for the e-learning system. Each level of the hierarchical architecture has an important role in generation of the courses, but it is the highest node that is working as a Learning Object Repository (LOR) for the system. There is no limitation for us in the number of Learning Object Repositories. In fact, the highest node is a shared space where several Learning Objects are available in there for professionals and learners. The idea of Learning Objects is grounded in the object-oriented concept of computer science. Object Oriented Programming (OOP) has employed for decreasing the cost of maintaining and producing large programs for computer science. The same as OOP, the principle of Learning Objects (LO) is the creation of didactic components that can be reused in variety of different learning contexts. Learning Object is designed for teaching a focused concept and it is relevant to the course content. It must have considerable contribution for the course subject. In fact, it is a fundamental reusable building block composed of all the didactically necessary components to comprise a self-contained instructional unit plus extracted metadata (that is a descriptor for the content and concept of the LO using for indexing). LO composed of multimedia learning materials and assets those are included, but are not limited to, Java applets, animations, tutorials, texts, Web sites, diagrams, audio and video clips, exams, pictures, maps, and assessments combined for the purpose of presenting interchangeable examples, arguments, cases, and practical exercises, which can be guided by teacher or based on a learner self determined need or adaptive, based on learner's talent and quality. In a distributed environment, Learning Objects vary in size, scope, and level of granularity ranging from a small set of instruction to a series of resources combined to provide a more complex learning package but the size of the objects must be tolerable for the distributed node. In other words, the size must be as small as possible for maintaining the response time of the relevant network. Course contents are generating with merging these Learning Objects by the content provider node. Without usage of Learning Objects' metadata, machine/human is not able to distinguish which is which among the large number of Learning Objects those are stored at the system. Learning Objects must be indexed in order to be interpretable for e-learning system and determine how they can be used during the learning process. This is done by an abstraction representation file (Metadata). A Metadata is a collection of attributes about a Learning Object (LO) describing some features such as its type (text, simulation, slide, questionnaire ...), the required educational level (high school, university ...), the language, the interactivity level and so on... Several organizations such as IEEE (IEEE Std 1484.12.1-2002), EDUCOM, IMS Global Learning Consortium, etc. are focusing their attention on the creation of Metadata standards specifying the syntax and the semantics of the attribute declarations. (Sangineto et al. 2007)

8. Multiple Author Course Generation

With current rapid changes in skills and knowledge and the advent of multidisciplinary educations, having more than one author for a course is normal. Moreover, in a multiple author course, learner has a great chance to study and compare variety of different point of views in a united package. For the sake of generating a course with homogenous content, there must be a good cooperation between authors and a unique rule and a coordinator (human or machine) for all of the authors for playing their role in the content generation. The role of coordinator is so important, and it must arrange proper authors for improving the quality of the course. Each author already has a local bank of information that is related to the subject of the course, and he/she needs to develop it before beginning to generate the content. Consequently, the course authoring flow is usually as the following steps:

1. Information aggregation and clustering;
2. Knowledge Object extraction;
3. Knowledge Object aggregation and clustering;
4. Creating and gathering training aid assets for the subject;
5. Merging training aids and knowledge objects to generate enriched knowledge objects;
6. Clustering the enriched knowledge objects and extracting its metadata;
7. Generation of standard learning object and its relevant metadata by merging enriched knowledge objects;
8. Making course and with merging learning objects;
9. Checking the correctness and quality of course by coordinator;

10. Voting for generated course from experienced reviewers and normal learners;
11. Revising the course with cooperation of authors by coordinator;
12. Go to step 1 for maintaining the course and keeping it up-to-date;

If Authors are human, these 12 steps will take long time from them, especially in first 7 steps. There is no specific limitation for information aggregation in step one. It is very hard for human to gather all the available information those are related to the subject. Knowledge extraction is relatively easier for human in comparison with a machine but the inference engine of human is not stable and her/his inference ability is changing from time to time especially when he/she encountered with huge volumes of data. Moreover, before execution of step two, the authors ought to read all the gathered information in step one that is a very time consuming task. Furthermore, authors ought to find relations among huge volumes of data those are needed for Knowledge Object extraction. This latter task is even harder than reading all the gathered information, especially in multi author course generation because in step 3 authors may encounter with large number of conflicted Knowledge Objects. Steps 4, 5, 6 and 7 are more reliable with human authors but expensive and time consuming. Therefore, nowadays, semi-automatic method is the best way for generating courses. Obviously, finding relevant data and merging them intelligently according to the didactic principles are very important in course generation. Next sub sections will try to describe all kinds of merging clearly without complicated mathematical formulas or program codes.

8.1. Conceptual merging of aggregated information

The first step in generating content is aggregation of relevant information. How we could find relevant information for our subject? Normally, we will search over local documents and then over the Internet for finding proper files, assets and documents. Information aggregator could be a Web agent that is running on a machine. It could be a number of human operators too. Nowadays, conceptual search engines are not ubiquitous, but a Web agent could download huge volume of data from World Wide Web, 24 hours per day, and 7 days per week using relevant keyword to the subject of the course content and a search engine. For example, if we search keyword "LOR" for our subject that is e-learning, we will find the following texts for its definition:

Definitions of LOR on the Web:
Lor is a commune in the Aisne department in Picardie in northern France.
Loricrin, also known as LOR, is a human gene.
The ratio of the total light output of the luminary to the output of the lamp(s), under stated conditions.
Learning Object Repository: A shared space where several learning objects are available to professionals and learners.
Letter of Recommendation
Level of Risk, i.e. low, moderate or high

Obviously, the fourth definition is the correct one but without conceptual clustering, the correct answer or relevant information is not detectable by machine. Someone must create a proper metadata that is understandable for a machine then it will be able to distinguish which one of the definitions is in scope of the subject. This will happen via using another level of abstraction that is Ontology. Ontology used to represent Domain Concepts and their relations (A Domain Concept is a concept belonging to the described instructional domain and could be explained by one or more Learning Objects (Sangineto et al. 2007)). The Ontology concept is not limited to this definition. In fact, it is applicable even for a primitive level of information such as a simple text paragraph. In this situation, Ontology could be used for representing the words and their relations. In this situation, the domain concept could be a simple didactic keyword. Thanks to very complicated Natural Language Processing tools those exist now, generation an enriched metadata for a piece of aggregated information (retrieved from local disk or downloaded from Internet) is possible and some programs for conceptual information clustering are under research and development. With the help of such programs, lowest nodes of the architecture those are depicted in figure 5 are able to gather and cluster the information that is relevant to the course subject in the line of making Knowledge Objects. Each Knowledge Object must have an attached metadata. The duty of this metadata is making the concept of Knowledge Object and its hidden pedagogical intention, interpretable for the machine as much as possible. (Zouaq et al. 2007) However, to do that, the node must be able to understand the concept of the course subject. Lots of solutions are representing for this problem. We have to feed the machine with some unique specification and relevant keywords of the subjects and ask it to find all matched information then machine must find the tiny didactic domains inside the gathered information via data mining algorithms and generate the Ontology for them. At the end of the process there must be an Ontology that is representing all didactic domain concepts and their relations. Each of these didactic domains or a set of them could be a Knowledge Object for us. The metadata of these Knowledge Objects will be generated based of the extracted Ontology.

8.2. Merging the aggregated information by reviewers vote.

As you can see in section 8.1 the search result for the definition of "LOR", only an intelligent engine that knows about the concept of e-learning via enriched metadata of the course could filter the useless information and refine information those are relevant to the subject. Obviously, a human that is familiar with the subject of the course (e.g. Course author or one of the course authors) could do the above duty at a glance. When the machine has found so many different correct parallel definitions from different resources for a specific concept, it is the human who can find which of the retrieved definitions is more complete so in this case merging by reviewers could be a good solution. In this method author could see the parallel definitions of a concept in a package and in front of each definition, there is the score of that definition that is generated by the vote of experienced reviewers. Obviously, author could use these voted definitions (those are confirmed by a number of experts in that field) for generating more reliable and confirmed knowledge objects.

8.3. Merging the aggregated information by authors

In this method, the refining and filtering of the gathered information is done by humans. This method is similar to conventional method of course authoring but there is a difference. In this method gathering of information is not done by the author necessarily. The author is just the one who uses the already gathered information for generating knowledge objects. These knowledge objects are based on the author point of view and responsibility. In case that there are multiple authors for the course, there must be a coordinator for auditing over the quality of the knowledge objects.

8.4. hybrid method for merging

In this method, the coordinator is taking advantage from all of the above methods. In other words, merging the human outcomes and machine outcomes will do by coordinator and he/she will use each method in its proper situation for obtaining the best quality in generated knowledge objects. In this method, the same as sections 8.2 and 8.3 the metadata must be generated by human that is the coordinator in this method.

9. Automatic Course Generation

In last section, we represent different ways for generating knowledge objects in a distributed environment. Knowledge Objects and their related metadata are the outcome of the activities those are described in last section. Clustering and aggregation of Knowledge Objects needs the same sequences those are described in last section for aggregating and clustering of information. One of the differences is that in addition to the processes those are described in last section, authors may have to create some multimedia assets and interactive animations or scripts for adding to the Knowledge Objects. Automatic generation of these assets is not easy for a machine and with current technology it is a task for human authors (see section 6). Merging these assets and Knowledge Objects will make enriched Knowledge Objects. This merging operation has to be done by the author and it is the duty of the coordinator in case that there are multiple authors for the course. There must be a metadata for each enriched Knowledge Objects. This metadata could be generated by machine or human but the final quality assurance must be done by the coordinator. The other difference is that there must be an instructional method for representing the course. How we can add this capability to the automatically generated course? In fact, there are some templates for learning theories and human operators could feed these templates to the machine.

Nowadays, some tools take an instructional theory as input (Zouaq et al. 2007) and offer a mapping of the theory's principles with the asset categories (instructional role Ontology). Instructional theory is a discipline that focuses on how to structure material for promoting the education of human beings and formalized as a set of Instructional steps those are expressed in the form of rules combining asset categories and predefined methods. Semantic Web Rule Language (SWRL) is a good tool for the rule formalism. SWRL could enable the hierarchical rules to be used with an Ontology Web Language (OWL) knowledge base. SWRL rules can then exploit OWL classes, instances and properties in their body and head. OWL is a knowledge representation language for authoring ontologies and is built on top of RDF (Resource Description Framework designed as a metadata model) and designed for being interpreted by machine not human that is using XML syntax. OWL is for modeling data and better than RDF for representing metadata.

9.1. Learning object generation from knowledge objects

The outcomes of the section 5 are enriched Knowledge Objects with their relevant metadata those are defined with OWL. In this section machine will try to make standard Learning

Objects (based on SCORM) from the already produced Knowledge Objects. The same as human author during writing a text book, machine needs some relevant notes, proper scientific background about the subject of the course and an instructional method for representing the course. Knowledge Objects (enriched or not) could play the role of the notes for machine. The extracted Ontology has the same role for machine as the scientific background has for human author and the instructional method must be defined for a machine based on advanced reliable learning theories as templates. Using these three, the machine could generate Learning Objects and their relevant metadata by ordering and merging proper enriched Knowledge Objects in a package. These packages must not have large volumes because the intrinsic limitations of the distributed environments in transferring data. As a rule of thumb, Learning Objects are not longer than 30 minutes. Obviously, the content of Learning Objects is depending on the instructional policy and templates those have feed to the content provider engine by the coordinator or author.

9.2. Course generation using conceptual LO aggregation

Using the enriched metadata of Learning Objects those are made from Knowledge Objects' metadata, the engine will try to merge the Learning Objects for making a concept. The merging process is guided by the instructional policy and templates those are feed by the coordinator to the content provider engine. For example, in the case that machine has provided Learning Objects those it takes 100 hours from learners to watch them, there may be a defined policy for the conceptual engine to select just 24 hours of Learning Objects those are enough for covering the course and in the case that 24 hours are not enough for covering all aspects of the subject, there may be an instructional rule to divide the course to some parts. These rules could be implemented using SWRL over OWL knowledge bases. How the tool could find that which set of the chosen Learning Objects from Learning Objects knowledge base has the proper coverage for the course?

The answer is inside the Ontology. The extracted Ontology that is described in section 8.1 is familiar with all the domain concepts those are relevant to the subject. Therefore, a set of Learning Objects has enough coverage for the course subject if all the relevant domain concepts have been described in the metadata of its related Learning Objects. The generated course must be equipped with metadata too. This metadata will help the e-learning engine to represent better service to learners, better coverage check and is very useful for updating processes.

9.3. Automatic creation of new courses

When the engine will receive a subject, it will try to make a course based on it by data mining over the aggregated information. The sequence of automatic course generation is similar to section 8.1 but the difference is that in this situation, we do not have any background science about the subject of the course and the only thing that we know is the exact and correct name of the subject. In fact, we need to an intelligent fuzzy method inside the content provider engine to find a set of keywords those are really relevant to the subject of the course. Afterwards these keywords will feed to the retriever Web agent and gathering of the related information will begin according to the sequences those are described later. In this kind of course generation human is just a policy maker and feed the instructional theory templates to the engine. Afterwards, he/she will check the generated course and doing some editing on it for maintaining the homogeneity with other courses. This kind of course generation is very useful for those researchers who are entering to a completely new field of science. Without this engine, they must spend too much for gathering the primary information while machine could do that for them when they are asleep.

9.4. Keeping generated course up to date

Keeping the generated course up with current technology is more important than generation of the course. There are lots of feedbacks from learners and lots of advancements in the area of the course. Therefore, updating the courses is a need. In regard to large number of courses in an e-learning system (for example, more than 1400 courses in Payame Noor University www.pnu.ac.ir), keeping them up-to-date with using human authors is very expensive. When course updating routine is scheduled, it will search over the Web and local free to use documents for finding new relevant domain concepts. These domain concepts will add to the set of the subject relevant domain concepts and then the Ontology must be updated based on newly added domain concepts. Afterwards, all the intelligent sequences will run again and new Learning Objects will be produced. The newly produced Learning Objects must be used to regenerating the courses. In this regeneration, the feedbacks of the learners for each Learning Object could be a useful guide and those Learning Objects, which have more negative feedbacks could be replacing by new ones. Obviously, the metadata of the course and its Ontology must also regenerate. The update sequences could run repeatedly or run by E-learning system administrator based on the owner of the system policy and the distributed network traffic limitations.

10. State of the Art

E-learning systems have been achieved to great advancements but still there are so many problems in the line of developing a course. Managers' demands for generating high quality courses at very short time are increasing day by day especially because the rapid change in the technology and the advent of new branches of science and technology. Therefore, non stop research on new aspect of course development is a need.

10.1. Making multimedia presentation from aggregated knowledge

With a complete library of characters, background and other animation primitives, making an intelligent game engine equipped with physics rules that can convert a text to animation is possible. Text to animation tools has been appeared in some patents and Chatbots (Alice Chabot) but they are at the very first steps. In fact, converting a text to animation is very similar to translation of the Chinese to English.

With using papers those are published in this field we could generate a super compiler for translating a text to a primitive flash based animation. The soul of idea is simple. The parser of the compiler must find the relation between words and generating the Ontology of the text. Afterwards the compiler must map each important actor in the text to one of the elements in its library. For example, a unique character in the animation library could be assigned to each subject in the sentence and there must be a good movement or state in the animation library for each verb in the sentences. Obviously, generating such a compiler is not so easy but the result is fantastic. With such a compiler, we could make an animation from a scenario at a very short time and it is a real advancement in generating high quality multimedia courses.

10.2. Conceptual merging of ready made pictures and movies

There are lots of free to use ready made relevant assets those are gathered by the retriever Web agents from local documents and Web or by human operators. In some cases analyzing these assets and making proper metadata for them could be a good task for machine. For example, there are tools for decompiling flash files those are able to convert flash files to lots of reusable objects, characters and primitive animations. Using the library that is generated by these tools could be very helpful in saving the time of course animator or the author. Moreover, with assigning proper metadata to each multimedia asset, the part of engine that described in section 5 could merge these assets and make a covered course from them if we already have a proper library of the multimedia assets.

10.3. making interactive intelligent humanoid interface for course presentation

An interactive intelligent human user interface is a good choice for course presenter. The idea that is behind Chatbots like Alice and Ellen could be very useful for this user interface. These Chatbots could represent the course by reading it and acting according to the text. Some deaf people could read from the lips of the Chatbot and the presence of a good looking Chatbot could prevent the learner to be bored during long hours of sitting behind the computer. Another application of these Chatbots is interviewing with learners to find their level of knowledge and guiding them to the courses at proper level according to the learner capabilities and interests.

11. References

Zouaq, A., Nkambou, R., Frasson, C. (2007). An integrated Approach for Automatic Aggregation of Learning Knowledge Objects. Interdisciplinary Journal of Knowledge and Learning Knowledge Objects. 3,(2007) , pp.135-162.

Sangineto, E., Capuano, N., Gaeto, M., Micarelli, A. (2007). Adaptive course generation through learning styles representation. Springer – Verlog. (30 October 2007). pp.1-23

Curbera, F., Duftler, M., Khalaf, R., Nagy, W., Mukhi, N. (2002). Unraveling the Web Services Web. IEEE Internet Computing. (Mar-Apr 2002).

Huhns, M.N. (2002). Agents as Web Services. IEEE Internet Computing. (Jul-Aug 2002).

Zheng, Q., Dong, B., Tian, F., Chen, W. (2008) A service-oriented approach to integration of e-learning Information and Resource Management Systems. 12th IEEE International Conference on Integration of e-learning Information and Resource Management System. Computer Supported Cooperative Work in Design, 2008. CSCWD 2008. (16-18 April 2008).

Chengh, H.K., Tang, Q.C., Zhao, J.L. (2006) Web Services and Service-Oriented Application Provisioning: Analytical Study of Application Service Strategies. IEEE Transaction on Engineering Management, Vol.53, No.4, (November 2006).

Fu, X., Bultan, T., Su, J. (2005). Synchronizability of Conversations among Web Services. IEEE Transaction on Software Engineering. Vol.31, No.12, (December 2005).

Vossen, G., WesterKamp, P. (2006). Towards the Next Generation of E-Learning Standards: SCORM for Service-Oriented Environments. ICALT 2006.

GuiLing, W., YuShun, L., ShengWen, Y., ChunYu, M., Jun, X., MeiLin, S. (2005). Service-Oriented Grid Architecture and Middleware Technologies for Collaborative E-Learning. The 2005 International Conference on Services Computing (SCC'05).

Navaro, L.I., Such, M.M., Martin, D.M., Peco, PP. (2006). Architecture Oriented towards the management of Learning Objects Repositories. ICALT'06.

Liao, C.J., Yang, F.C.O. (2004). A Workflow Framework for Pervasive Learning Objects Composition by Employing Grid Services Flow Language. ICALT 2004.

Li, S.T., H.Lin, C. (2005). On the Distributed Management of SCORM compliant Course Contents. AINA'05.

Madhour, H., Sfaxi, M.A., Forte, M.W. (2006). Ownership Detection and Protection for Learning Object. IEEE International Conference on Advanced Learning Technologies (ICALT'06).

Frosch-Wilke, D. (2004). An Extended and Adaptable Information Model for Learning Objects. IEEE International Conference on Advanced Learning Technologies (ICALT'04).

Shih, W.C., Yang, C.T., Chen, P.I., Tseng, S.S. (2006) Performance-Based Content Retrieval for Learning Object Repositories on Grid Environments. The Seventh International Conference on Parallel and Distributed Computing, Application and Technologies (PDCAT'06).

Bahrami, K., Abedi, M., Daemi, B. (2007). Modeling the 3-Tier E-learning system on power line network. ICWS 2007. (2007) pp. 888-895

Bahrami, K., Abedi, M., Daemi, B. (2007). Implementation of Distributed E-learning system on Power Line Network. The third Advanced International Conference on Telecommunications. AICT 2007. (2007).

Bahrami, K., Abedi, M., Daemi, B. (2007). A Web service based Portal framework for distance learning on Power Line Network. Sixth International Conference on Information, Communication and Signal Processing. ICICS 2007. (December, 2007)

E-learning

Edited by Maja Jakobovic

ISBN 978-953-7619-95-4
Hard cover, 312 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

E-Learning is a vast and complex research topic that poses many challenges in every aspect: educational and pedagogical strategies and techniques and the tools for achieving them; usability, accessibility and user interface design; knowledge sharing and collaborative environments; technologies, architectures, and protocols; user activity monitoring, assessment and evaluation; experiences, case studies and more. This book's authors come from all over the world; their ideas, studies, findings and experiences are a valuable contribution to enriching our knowledge in the field of eLearning. The book is divided into three sections. The first covers architectures and environments for eLearning, while the second part presents research on user interaction and technologies for building usable eLearning environments, which are the basis for realizing educational and pedagogical aims, and the final last part illustrates applications, laboratories, and experiences.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821