

A MULTI-AGENT ENABLED E-EDUCATION OBJECT & COURSE AUTHORIZING SYSTEM (MEEOCAS)

Anbo Meng, Luqing Ye, Romuald Stock, Pierre Padilla

ABSTRACT

The paradigm of developing on-line courses by assembling existing e-education objects (EEO), stored in databases, is increasingly revealing its advantage and attracting more and more attention of researchers and practitioners in the e-learning domain. However, there is still not an acknowledged definition: what constitutes an EEO or what size such an EEO should be to maximize its reusability and flexibility capacity. In this paper, firstly, the design philosophy of MEEOCAS was identified: All the roles (i.e., Learners, course /EEO designers and teachers using MEEOCAS) are both consumers and producers who do not only benefit from but also contribute to this system. Based on the principle mentioned above, an abstract conceptual model of EEOCAS was designed. It provides an efficient and powerful mechanism to facilitate the developing process of courses and learning objects by virtue of the incorporation of software agents and learning object technologies. Secondly, for our purpose, a new definition of learning object—EEO, which stems from the idea of Object-Oriented Programming (OOP) and the model of Cisco’s reusable learning object (RLO), was put forward. Furthermore, an XML based EEO packaging model was also described. Thirdly, the implementation model of MEEOCAS based on JADE (Java Agent DEvelopment Framework) platform fully complying with the FIPA specifications was presented. Relying on the common platform, all the involved agents can conveniently communicate, collaborate, and negotiate with each other, in order to perform some specified tasks using the common domain ontology and XML content language. Finally, some application scenarios based on UML schema, such as searching the appropriate EEOs, subscription to LMMA and the negotiation between course and EEO developers, were demonstrated. In conclusion, the aim of the MEEOCAS project is to make easy the development and deployment of learning contents and to build a self-improving and increasingly accumulated e-education resource library.

KEYWORDS

E-education object, multi-agent, XML, JADE

INTRODUCTION

As is known that “content is king”, therefore, as far as e-learning is concerned, one key issue is how to develop instructional materials of high quality that could be reused and applied to different contexts. Unfortunately, these instructional contents are, traditionally, expensive and time-consuming to produce. In most situations, course authors have to create their new course from the scratch, even though numerous referenced instructional materials are conveniently accessible; course developers (authors/teachers) have to break them down into smaller constituent components at the beginning, and then modify or reassemble them in their own way that supports their individual instructional goals. Such repeated creating process, undoubtedly, is toilsome, tedious and time-consuming. The more important point is the difficulty in sharing and reusing of these courses even when they are available, how to share and reuse them is still a big problem. Since the traditional courses generally fixed in length, sequence, and scope, are built as such a large monolithic structure that it is difficult to re-purpose them into other contexts at such a course-grained level via Internet. In fact, this large and inflexible structure misses out on most of the benefits to authors and learners.

Fortunately, the recent emergence of learning object seems to be a promising solution to the problem due to the potential of its reusability, interoperability, adaptability, and scalability. However, another issue arises when taking into consideration how to apply these ready-made learning objects to course authoring system and deploy them to the learning process in an efficient and flexible manner. To address this issue, the agent technology has been introduced in MEEOCAS. The combination of learning object and agent technology makes it possible for MEEOCAS to facilitate the development of course and learning object itself, and furthermore, to personalize learner's learning process and patterns.

DESIGN PRINCIPLE AND CONCEPT MODEL OF MEEOCAS

When considering the design of MEEOCAS, to overcome the inflexible monolithic structure of traditional course and the low efficiency of creating process, as well as, to comply with the functional requirements of different occasions, our philosophy is: *Learners, course materials designers and teachers are both consumers and producers when using MEEOCAS.* That is to say, learners, course materials designers (i.e. learning object designers and course designers) and teachers not only directly or indirectly benefit from MEEOCAS but also contribute to MEEOCAS and thus a self-improvement and increasingly accumulated learning resource library can probably be worked up. Figure 1 is the concept model of MEEOCAS, in which, we introduced the well-known concept of *learning object* as supporting groundwork, which can be re-purposed for many use and we emphasize more on the performance support than simply an information system composed of learning objects. From learning activity-enabled perspective, learning object will play a key role in several aspects in MEEOCAS (see figure 1) as follows:

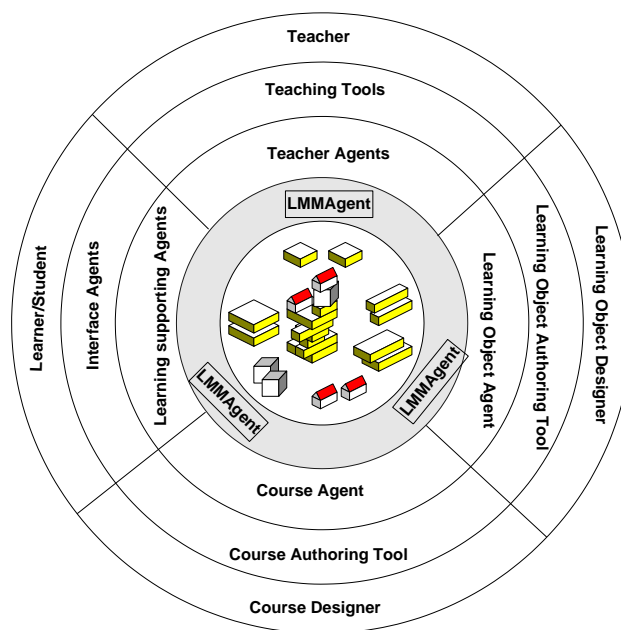


Figure 1. Concept model of MEEOCAS

- Facilitating the development of courseware, thanks to the reusability, deliverability and discoverability of learning object, mainly through the support by Course Assistant Agent (CAA) in cooperation with both Learning Material Management Agent (LMMA) and Object Learning Assistant Agent (OLAA).
- Facilitating the development of learning object in virtue of its modifiability, inheritability, and discoverability etc. This process is chiefly assisted by OLAA and LMMA.
- Providing learners with personalized learning experiences and tailored learning services through intelligent navigation and dynamic learning paths modification according to their own learning performances and profiles in automatic tutoring mode, which is supported by the cooperation and coordination between Pedagogic Agent (PA), Learner Model Agent (LMA), and Evaluation Agent (EA) etc.

- Teachers can gain access to learning object libraries and make full use of them in teacher-centered learning mode with support from Teacher Assistant Agent (TAA).

DEFINITION OF LEARNING OBJECT IN MEEOCAS

In the e-education context, the emergence of learning object is assuredly exciting and encouraging. As [Wiley 2000] says Reusable Learning Objects (RLOs) are emerging as the “technology of choice in the next generation of instructional design, development, and delivery, due to its potential for reusability, generativity, adaptability, and scalability.” It is certain that having a library of learning objects to draw on will sharply shorten the course development time when allowing for faster deployment of the learning and personalize the learning process.

However, learning object is a relatively new term, thus we are not surprised to find numerous versions of different definitions of learning object: Learning Object Metadata (2001)-IEEE 1484.12.1 defines a learning object as “any entity, digital or not-digital, which can be used, re-used or referenced during technology supported learning”. Cisco (2001) defines a learning object as “a granular, reusable chunk of information that is media independent”. Wiley (2001) defines learning objects as “elements of a new type of computer-based instruction grounded in the object-oriented paradigm of computer science”.

Besides various definitions of learning objects, there are also a large amount of terms relevant to it, like “Reusable Learning Object (RLO)”, “Reusable Information Object (RIO)” (Cisco 2001), “Assignable unit”, “Sharable content object (SCO)” (ADL SCORM 2001), “knowledge objects” (Merrill, Li, & Jones 1991), “online learning materials” (MERLOT 2000), “educational software components” (ESCOT 2001) and etc. This paper uses the term E-education Object (EEO) to describe its purpose and functionalities. Here gives the definition of EEO conforming to the functionalities requirement of learning object in MEEOCAS: *An E-education Object (EEO) is a reusable, modifiable, scalable, inheritable, polymorphous and multipurpose component that encapsulates well-organized “raw assets” (i.e. contents, practices and assessments) as well as a common interface attached to it.*

This definition is derived from some concepts of object-oriented programming, and also references to the models of ADL’s SCORM and Cisco’s RLO. It is found to be an effective way to describe and construct an EEO. The more detailed interpretation is given as follows:

- Reusability indicates that an EEO can be reused in certain context at random times without any modification.
- Inheritability implies that an EEO can be entirely or partially (contents, practices or assessments) inherited by other ones. By this means, a wonderful experience of creating a new EEO applied to other contexts can be gained.
- Modifiability denotes that an EEO can be modified and then form a new EEO, e.g. when an EEO becomes out of date or not appropriate for most learners, this EEO should be updated, modified or even deleted.
- Scalability refers to the granularity of a learning object that can range from as small as a section to as large as a lesson. (Note: in MEEOCASS, the hierarchy of a course is like this: course->unit->lesson->section. In principal, the largest learning object is constrained within a lesson consisting of units, several of lessons constitute a course intended to deliver in MEEOCAS to learners for accomplishing their knowledge, skill, competence etc.)
- Multipurpose implies that an EEO can be applied to several contexts. E.g. in MEEOCAS, an EEO can be used by several actors/participants such as learners, teachers, EEO designers, Course designers and related agents.
- Polymorphous implies that the same subject matter may possibly own several versions of representing forms, e.g. simulations, demonstration, experiments, animations, html, text, video clip and games etc, which are likely to point to a single learning objective in order to accommodate to different learning style.
- EEO is a structural component that consists of well-organized “raw assets” such as content, practice and assessment.

- EEO has a common interface which makes it possible to be accessed, discovered and connected to the outside world.

Of course, the issue of intellectual property has to be taken into account when an EEO needs to be modified or reused by others rather than the original creator, an imaginable approach to this problem is to get the permission of the original copyright holder or else pay for the reuse of it.

STRUCTURAL MODEL OF EEO IN MEEOCAS

Firstly, we introduce and discuss an influential model—Cisco Reusable Learning Object (RLO) Model and then present a more flexible and practical structural model of EEO on the basis of it.

As a worldwide leader in networking for the Internet and one of the forerunners in learning object design, creation, and deployment, Cisco is also actively participating in standards groups such as IMS and ADL, whose RLO strategy has been attracting extensive attention all the while. As such, it is worth taking a look at their RLO structure. The Cisco RLO is created by combining an overview, a summary, and from five to nine (7±2) reusable information objects (RIO) (see figure 2). Each RIO is built upon a single objective. Several RIOs are combined together to create a Reusable Learning Object (RLO). If a RIO can be equated with an individual component of a learning objective, a RLO is the sum of RIOs needed to fulfill that objective. To aid in content standardization, each RIO is further classified as concept, fact, procedure, process, or principle. Each of these RIO types has a recommended template that authors can follow to build the RIO.

O V E R V I E W	Content	Content	Content	Content	Content	S U M M A R Y
	Practice	Practice	Practice	Practice	Practice	
	Assess	Assess	Assess	Assess	Assess	

Figure 2. Cisco’s RLO and RIO
(Note: each column represents a single RIO)

As above-mentioned, the reusability of Cisco’s RLO model occurs at the level of RIO. Its internal elements (i.e. content items, practices items and assessment) cannot be reused in other contexts. Consequently, it is inconvenient and inflexible in the situation in which we only need reuse or inherit part of a RIO. According to ADL SCORM, any deliverable “raw medias”, such as illustrations, documents or media streams, can be seen as “assets” ready to be reused in other “content object”. Although a single “asset” cannot be used as a learning object alone, multiple learning objects can reuse these assets for gaining their objective. Consequently, from our standpoint, each content, practice or assessment in Cisco’s RIO is also a valuable asset which seems to be a more “meaningful chunk” than “raw media” mentioned in ADL SCORM.

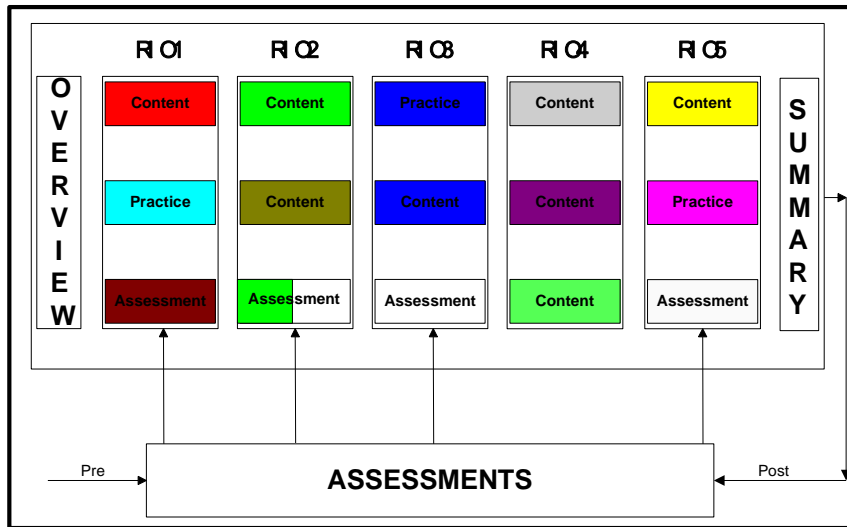


Figure 3. Recommended E-education Object (EEO) Modal

From figure 3, we can see that there are mainly two differences between the Cisco RLO model and the recommended EEO model: Firstly, the recommended model still contains several RIOs, but the structure of each RIO included in an EEO is not exactly the same. In Cisco RLO model, and yet all RIOs are as like as two peas, i.e. the internal structure and sequence (content-> practice ->assessment) of each RIO is unchangeable. In recommended EEO model, however, the inside of different RIOs may manifest different structures and sequences. An example is that, in RIO3, the component “practice” can possibly appear before the component “content” for some specific need. In another example, RIO4 may only contain the elements “content”. The second and also the most different point is that in the recommended EEO model, all the “assets” such as contents, practices or assessments in an EEO can be entirely or partially reused /inherited by other EEO. To be more exactly, all the boxes of different colors representing different assets in figure 3 can possibly be reused in other contexts. An example is used here to illustrate this idea. In Figure 4, assuming that an EEO designer is creating a new EEO3, like a course designer does, it is not necessary for him/her to develop this EEO from the scratch since there is a library of existing EEOs “waiting” for reuse. So the creation of EEO3 is relatively an easy and comforting process.

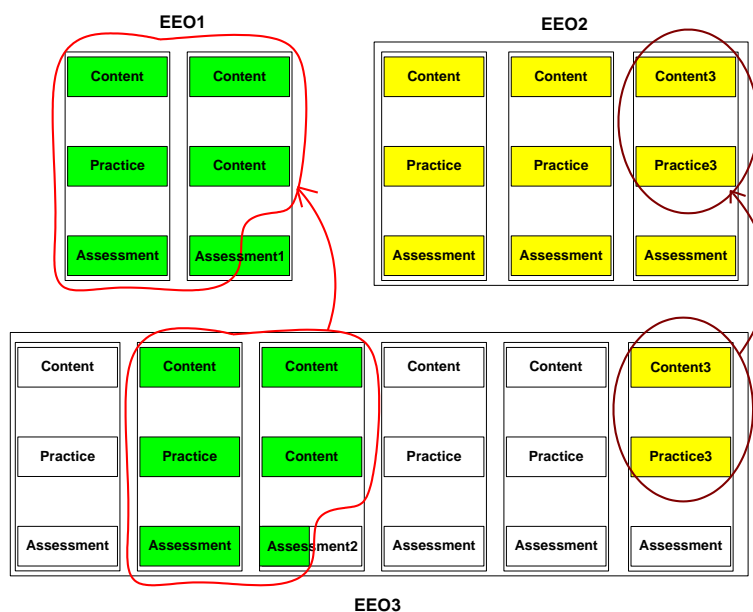


Figure 4. An example of demonstrating how to inherit assets from existing EEOs

As you see in figure 4, the accomplished EEO3 has inherited content3 and practice3 from EEO2 and furthermore almost all the assets in EEO1 with the exception of component “Assessment1” that has modified into Assessment2 in EEO3. Of course, the remainder represented by white boxes is created by the EEO designer himself/herself.

Using the recommended EEO model can bring us many benefits: On the one hand, it provides more flexibility and greater return on investment from the perspective of reusability and commerce. For example, in learning object-oriented learning pattern, a learner can make use of an EEO as a stand-alone performance support tool. The EEO gives learners the learning context, knowledge and skills needed to perform the given objective, and a method to assess mastery. EEOs and RIOs can also appear as offerings on a “road map” that is customized to the needs of each learner. Learners can see from this road map what they need to take, what they have completed, and what their learning destination is.

On the other hand, we can adequately “borrow” from Cisco other advanced ideas in building a learning object, such as its strategies of taxonomy and sequencing. What excites us most is that, based on our multi-agent system, a flexible, personalized and dynamical e-education environment can be formed because of the “participation” of EEO.

PACKAGE MODEL OF EEO

After an EEO is well accomplished, it has to be packaged and stored into database so that it can be reusable and accessible. When considering encapsulating an EEO into the package, it is a good practice for us to comply with some international specifications or standards (e.g. the specifications and standards of IMS, IEEE and SCORM). In the package, the key is to clearly describe the structure of an EEO and its corresponding resource. Fortunately, metadata has this ability to provide a common means to describe things so that EEOs can be self-describing and can be searched, found and applied to a specific context. Here gives a recommended package model of EEO (see figure 5), which includes: manifest XML, EEO metadata XML, RIO metadata XML, Raw asset metadata XML, Physical files and Package interchange file (PIF). Their functions are described as follows:

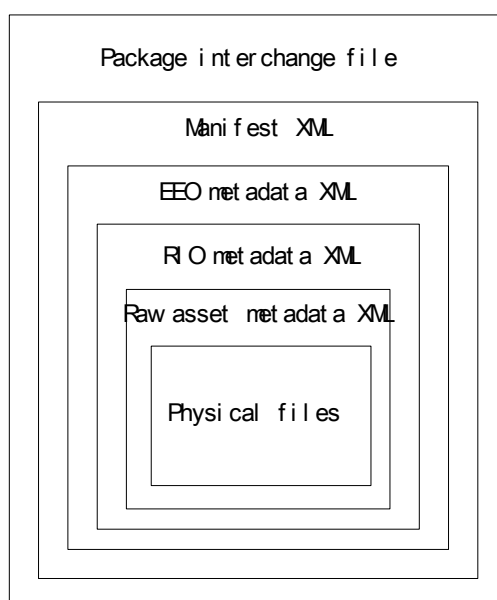


Figure 5. A recommended EEO package model

- PIF: it can be a zip file or other archive format allowing for standalone external use.
- Manifest XML file: it describes the file list needed when using the EEO.

- EEO metadata XML file: it describes the structure and other metadata information such as General, Lifecycle, Metametadata, Technical, Education, Relation and Classification etc.
- RIO metadata XML files: it describes the content construct.
- Raw asset files: it describes itself.
- Physical files: The physical files are the actual resource files.

ARCHITECTURE OF MEEOCAS BASED ON JADE

As mentioned above, it is sure that having a library of read-made EEOs to draw on will sharply shorten the course developing time. However, when allowing for how to make them to be deployed conveniently in a system and to be found easier by EEO and course developers, obviously, we still need a common communication environment to support it. Naturally, we introduce multi-agent technology to realize our system—MEEOCAS, which is based on Java Agent Development Environment (JADE 2003). JADE is a software framework fully implemented in Java language. It simplifies the implementation of multi-agent systems through a middle-ware that claims to comply with FIPA specifications and through a set of tools that support the debugging and deployment phase. The agent platform can be distributed across network and the configuration can be controlled via a remote GUI. According to our experience of using JADE, it justifies this feasibility of building a flexible and powerful MEEOCAS (see figure 6) with JADE.

From figure 6, we can see that learning material-side agents reside on server-side. However, the course interface agent, course assistant agent, EEO interface agent and EEO assistant agent can move to or be downloaded on the client-side so that all the EEO and course authors can conveniently develop their EEOs or courses at dispersed places. It is worth of note that MEEOCAS is a peer-to-peer distributed system that provides a common channel for communication, so the boundary of server side and client side has become somewhat blurry in most cases. The following is their functionality description of all involved agent in MEEOCAS and their further applications will be presented in the next section.

- *Course interface agent (CIA)*
CIA serves as a GUI of the course author. CIA provides a template-based course-authoring tool that facilitates developing consistent courses composed of EEOs. Besides, as an autonomous agent, CIA has all the characteristics of an agent.
- *Course assistant agent (CAA)*
CAA helps course author perform many specific tasks such as searching existing EEOs, subscribing services (e.g., EEO) to LCMA, negotiating with other EEO authors, etc.
- *EEO interface agent (EEOIA)*
Like CIA, EEOIA provides an EEO development environment facilitating EEO authors to develop consistent EEO/RIO
- *EEO assistant agent (EEOAA)*
Like CAA, EEOAA performs many specific tasks such as searching existing EEOs, subscribing services (e.g., EEO) to LCMA, negotiating with other course authors, etc.
- *EEO provider agent (EEOPA)*
EEOPA is responsible for the communication with EEO repositories/databases
- *Course provider agent (CPA)*
CPA is responsible for the communication with course repositories/databases
- *Learning material management agent (LCMA)*
LCMA is responsible for the management of life cycle of all involved agent in MEEOCAS such as the creation, deletion, suspension, resumption, authentication and migration of agents. LCMA also provides the yellow page service. For example, when an EEO accomplished, it can be registered with LCMA agent so that course author can find it.

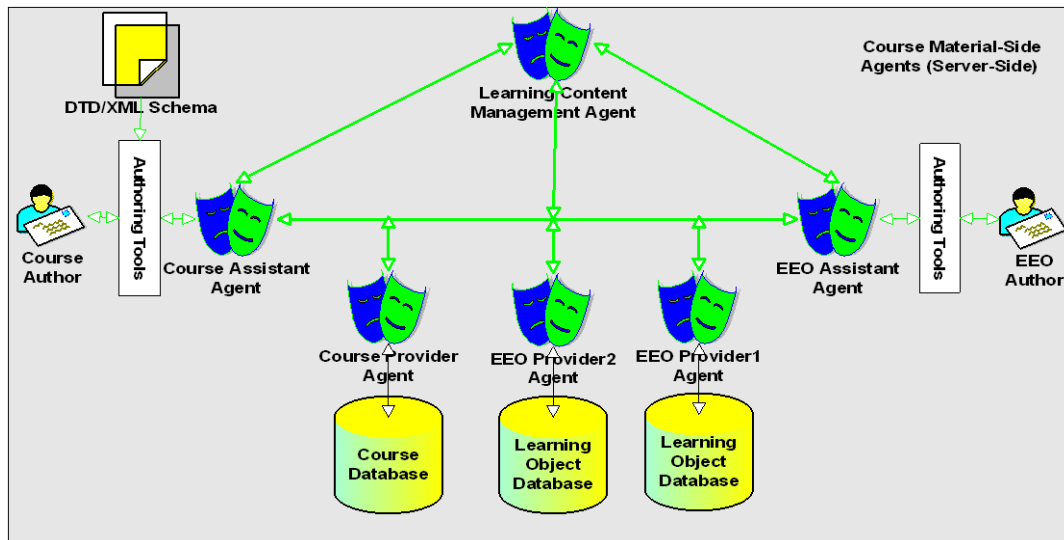


Figure 6. Architecture of agent enabled course authoring model based on learning object

APPLICATION SCENARIOS

With MEEOCAS, several application scenarios can be easily applied, such as searching the appropriate EEOs, the subscription to LCMA and the negotiation between course and EEO developers etc. To save spaces, only one UML sequence schema, used to demonstrate the conversation protocols when searching an EEO, is given here. Figure 7 shows that course developers can search the EEOs that he/she prefers and subscribe to the LCMA when the search fails. These agents involved include CIA, CAA, LCMA, EEOPA and Federal LCMA. The detailed interaction steps are shown below.

- **Step 1:** course author states desired EEO as a goal to course interface agent; CIA may provide a particular form consisting of learning object metadata (LOM) to facilitate the course developer to customize his/her desired EEO (e.g., if the course developer needs such an EEO as Context='continuous formation', Interactive Level='middle', Difficulty='high', Keyword='agent, communication', Language='French', etc., s/he may fill out the ready-made form compliant to the IEEE LOM standard (2002) to customize his/her required EEO).
- **Step 2:** CIA sends an ACL message to CAA (note: the ontology of the ACL message is called 'EEO LOM Ontology' in MEEOCAS, which, in fact, is the LOM schema that represents the element concepts and their relations of LOM, and its content codec language may be XML or SL language).
- **Step 3:** CAA responses to CIA, which refers to the acceptance of CIA's request
- **Step 4:** CAA asks for help from LCMA that can identify whether there exist EEO provider agents providing such kind of service as described in the ACL message of CAA (note: in MEEOCAS, this CAA message applies a template to searching such service). In MEEOCAS, there maybe exist several LCMA agents that compose the federation enabling the flexibility and distribution data storage. As such, when a local LCMA cannot find appropriate EEO providers, it can deliver the CAA's request to other federal LCMAs for help.
- **Step 5:** LCMA returns a list of names of agents that match the template, if no match is satisfied, LCMA sends a message inquiring whether CAA is willing to search further.
- **Step 6:** CAA agrees with LCMA.
- **Step 7:** LCMA asks for help from the federal LMCA agents
- **Step 8:** Implying that the federal LMCA agent has found the intended EEO provider.
- **Step 9:** According to the returned list of provider agents from federal LCMA, CAA sends corresponding ACL messages to all of the EEO provider agents, which first match the request with their own EEO metadata info. If these matches are satisfied, the locators (e.g., a URL or URI pointing to the desired EEO) will be sent to CAA. Or else, failed message may be presented to the course developer and CIA may inquire further whether the course developer is willing to modify his/her form or subscribe to such service from LCMA which will automatically notify CAA as soon

as such service is available in MEEOCAS.

- **Step 10 and step 11:** Course assistant informs CIA to present the desired EEO with a XML type sheet to which the course developer prefers.
- **Step 12:** Once having accomplished a courseware by taking advantage of the template-based course authoring tools, course developer can store his/her works into the course repository. This process is performed through **step 13 and step 14.**
- **Step 15:** When a new course is stored into the course repository/database, course provider agent will register the new service to LCMA agent.
- **Step 16 and step 17:** Return the message of success registration.

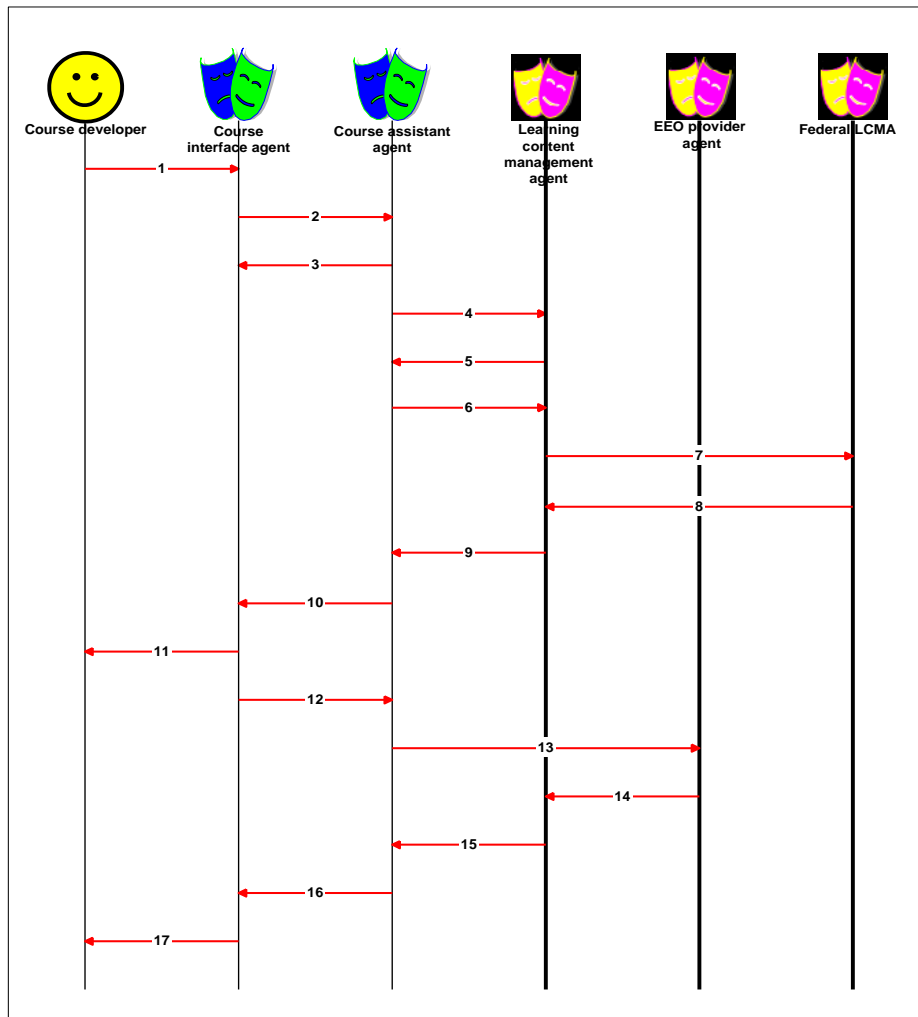


Figure 7. UML sequence schema of searching target EEO

CONCLUSIONS

This paper presented the architecture of a multi-agent based learning object and course authoring system. With the introduction of EEO into MEEOCAS, we can find out that it brings many advantages as follows:

- Course designers may conveniently develop their courses through assembling the ready-made EEOs instead of creating them from the scratch. It is sure that, when contributing courses to MEEOCAS, they could also make full use of the existing courses which contain well-organized and well-tagged EEOs provided by MEEOCAS.
- From the standpoint of EEO designers, it is convenient for them to create EEOs adapting to corresponding templates in accordance with an appropriate taxonomy, and to update the old version

of EEOs according to the feedbacks from all the other actors (e.g. teachers, course designers, learners or the agents in MEEOCAS). In addition, EEO designers can also formulate different EEOs satisfying diversified levels of learning objectives by inheriting, modifying and reassembling the existing EEOs.

- From the perspective of teachers, they could take advantage of these ready-made EEOs and courses to organize their teaching process or offer learners some recommendations of useful learning recourses during their interactive pedagogic activities.
- As far as learners are concerned, in MEEOCAS-enabled environment, they may choose among several available learning patterns (e.g. course-oriented learning, EEO-oriented learning, self-paced learning, teacher-centered learning or collective learning pattern etc.). In converse, the feedbacks from learners are also important references for course and EEO designer to modify and improve their works.

Despite these potential advantages of EEO, it is still difficult to make them function without a flexible supporting environment. Just because of the incorporation of agent technology in MEEOCAS, which makes possible to provide a common channel for the communication, cooperation and negotiation among all the agents. As such, it is convenient to deliver and handle EEOs throughout MEEOCAS with the help of relevant agents. For MEEOCAS, it not only facilitates the development of courses and EEOs, but also ~~s~~, when applying MEEOCAS ~~applies~~ to the whole e-education system, makes easy to personalize the learner's learning process and diversify their learning styles ~~because of~~because of the wonderful features of EEO and agent. With the advancement of multi-agent and Internet technology, ~~we~~ it is believed that ~~more and more~~ similar systems dedicating to the development of e-education resource library will ~~surely~~surely be emerged more and more.

REFERENCES

ADL SCORM. (2001). Advanced Distributed Learning Initiative: Shareable Courseware Object Reference Model. Available at: <http://grouper.ieee.org/ltsc/meeting/199912/doc/sco-km>

Cisco. (2001). Elearning glossary. Available at: <http://www.cisco.com/warp/public/10/wwtraining>

Cisco Systems, Inc. (2001). Reusable Learning Object Strategy v4. US: Author. Available at: <http://www.Cisco.com>.

Fabio Bellifemine, Giovanni Caire, Tiziana Trucco. JADE (2004) programmer's guide. Available at: <http://jade.tilab.com/>

Learning Object Metadata (2001) EEE 1484.12.1. WG12: Learning Object Metadata. Available at: <http://ltsc.ieee.org/wg12/>

Merrill, M.D. (2000). Knowledge objects and mental models. In D.A. Wiley (Ed.) The Instructional Use of Learning Objects. Available at: <http://reusability.org/read/chapters/merrill.doc>.

South, J. B., & Monson, D. W. (2001). A university-wide system for creating, capturing, and delivering learning objects. In D. A. Wiley (Ed.), The instructional use of learning objects. Available at: <http://reusability.org/read/chapters/south.doc>.

Wiley, D. A. (2001). Connecting learning objects to instructional design theory: a definition, a metaphor, and taxonomy. In: D. A. Wiley (Ed.), The instructional use of learning objects. Available at: <http://reusability.org/read/chapters/wiley.doc>.

Anbo MENG
Huazhong University of Science of Technology,
Wuhan 430074, China
Email: menganbo@vip.sina.com
a.meng@enim.fr