

Retrieving Learning Resources over the Cloud

Fernando De la Prieta¹, Juan F. De Paz Santana¹, Ana B. Gil¹,
and Magali Boureux²

¹ Department of Computer Science and Automation, University of Salamanca
Plaza de la merced s/n, 37008 – Salamanca, Spain

² Dip. di Filosofia Pedagogia e Psicologia
Università degli Studi di Verona, Lungadige Porta Vittoria 17, 37129 Verona, Italy
{fer, fcofds, abg}@usal.es, magali.boureux@univr.it

Abstract. Reusing resources has been on the rise recently in the ICT sector. In fact, this trend is expanding into other areas such as the educational sector. Learning objects have made it possible to create digital resources that can be reused in various didactic units. These resources are stored in repositories, and thus require a search process that allows them to be located and retrieved. The present study proposes the AIREH tool, which was deployed into a cloud environment and facilitates the retrieval of learning objects by integrating virtual organizations and agents with CBR systems that implement collaborative filtering techniques.

Keywords: Learning objects, e-learning, cloud computing, collaborative filtering.

1 Introduction

Recent years have seen the rapid evolution, essentially a revolution, in methods for creating, updating and packaging digital resources oriented to education [12]. As a result, it is now possible to create new educational experiences by developing self-contained educational units. Each of these modular components is known as a Learning Object (LO) [9]. It is necessary to create systems and procedures for storing and retrieving LOs in a way that allows the content to be easily reused.

A LO can be described as a digital, self-contained and reusable entity with a clear educational purpose, composed of at least three internal components: content, learning activities, and contextualization elements. Additionally, they must include an external structure that facilitates their identification, storage and retrieval in metadata format [5]. The advantages of this new technology seem to be evident [5]: reuse, personalization, durability, granularity, flexibility and accessibility of educational resources, all of which involve twice the savings in cost and time for preparing complete educational activities. However, there are also disadvantages [13], primarily the period of adaption required by educators to a new educational context in which new technologies are coupled with innovative methodologies for creating and publishing didactic resources. This can be considered a new paradigm in the educational sector.

This new paradigm initiates a new set of technological goals related to the new life-cycle of creating educational experiences. Among the most important of these

are, first of all, the creation of packets with educational content, a situation that seems to have been overcome [12] by the establishment of international standards for LO [7], and other standards such as SCORM (Sharable Content Object Reference Model) [10]. Secondly, the publication of these educational resources, which is being achieved through LO Repositories (LOR). Despite their overly heterogeneous technology [14], LORs make it possible to access resources through standardized query languages [8]. Finally, the need for tools that can quickly, simply and efficiently search and recover LOs from repositories.

This study presents the AIREH tool (*Architecture for Intelligent Recovery of Educational content in Heterogeneous Environments*) [18], which makes it possible to search and recover educational resources encapsulated in the form of a LO. Similarly, a system can use a CBR (Case-Based Reasoning) system to recommend which educational resources might be of particular interest to the user, based on information from previous searches. This system is based on Multi-Agent Systems (MAS) based on virtual organizations (VO). Finally, it should be noted that this application was deployed in a Cloud Computing environment, which allows users to store information about the recovered resources in a cloud.

This study is organized as follows, next section establishes the state of the and the related work, section 3 shows the proposal system, section 4 the experiments and the case study and the last sections contains the conclusions and future work.

2 State of the Art and Related Work

2.1 Educational Technology

The concept of LO has become a central component within the new technological paradigm in an educational context. Revolving around the concept of LO are such relevant elements as LMS (Learning Management System), LOR, authoring tools, or systems for discovering didactic resources, among others. The IEEE's Learning Technology Standards Committee (LTSC) defines a LO in general terms as *any entity, digital or non-digital, which can be used, re-used or referenced during technology supported learning*. In short, practically any educational resource can be considered a LO, a fact heavily criticized by various authors [5][8][9] who have tried to delimit the concept as much as possible.

What seems perfectly clear is that any LO must be associated with an external structure that facilitates its search, evaluation, recovery and eventual reuse. There are currently different standards and specifications specifically designed to describe educational resources by means through the use of metadata. The most relevant standards are Dublin Core [6], which is more oriented towards digital resources in general and is quite widespread within a library context; IEEE LOM (Learning Object Metadata) [7], which is the most commonly used standard for describing LOs; and finally SCORM [10], which is oriented to packaging and distributing complete educational activities.

LOs are commonly stored in repositories, which are characterized by their heterogeneity [17], including different storage systems, access to objects, query

methods, etc. The heterogeneity itself is not a problem, since there are different tools that can isolate the internal logic of the LOR from the exterior, which in fact makes it possible to automatically search different repositories simultaneously using a single query application. Most notable among these tools, which serve as a middleware layer between the repository and the clients, are SQI (Simple query Interface)[15], which was standardized by the ECS (European Committee for Standardization), and OAI-MPH (Open Archives Initiative Protocol for Metadata Harvesting) [16]. The basic function of these systems is, at a conceptual level, trivial, being based on web services, which allows the client to perform a query in a LOR, which in turn provides results in the form of LOs. The importance of this type of query method stems from the fact that it is not limited to a particular query language or format in which the query results are returned, nor are the internal characteristics of the server restricted.

LOR are currently classified according to their topological distribution as follows:

- **Autonomous repositories.** Although the opposite would seem to be true, these are currently the most widespread repositories. They do not have a system that allows an external search, and thus require searches in situ, normally using a web interface.
- **Middleware repositories [11].** They have an external search interface and may include an automated search system.
- **Federated search systems [4].** These systems, including the one used for the present study, perform automatic searches in other repositories, but do not store LOs or educational resources.
- **Repositories with federated search [11].** In addition to performing internal searches, they also perform automated searches in other repositories.

2.2 *Cloud Computing*

Cloud computing uses a new technology that develops applications in a way that allows both the execution of the application and the storage of data to be performed ubiquitously for all users. Cloud architecture provides user support at different levels that vary according to their different characteristics. The different levels can be described as follows [2]:

- **Infrastructure as a Service (IaaS):** remote management and control of hardware resources provided by a system.
- **Platform as a Service (PaaS):** offers the cloud platform along with a series of libraries to develop applications in which the distribution of tasks, the persistence and other layers are transparent for the developer.
- **Software as a Service (SaaS):** consists in offering different applications to be used through the internet as opposed to a local installation.

There are incipient developments that broach the topic of cloud technology and e-learning [1], however few studies incorporate both concepts. As demonstrated in

[1], cloud computing and e-learning are fundamentally centered on the SaaS layer. Other research such as [2] has studied the applicability in different fields such as education, but also focuses on the SaaS layer, ubiquitously offering a set of applications to users. Due to these circumstances, and given the possibilities offered by cloud computing in the field of e-learning for managing applications in the SaaS layer, it has become necessary to develop applications in cloud architectures that can form part of the PaaS layer, and that can manage the LOs that are stored within the architecture, thus facilitating the interaction and access of the applications developed in the system.

2.3 Related Work

The primary goal of search systems is, unquestionably, to provide users with the most appropriate LO, not just for a particular search, but for their general interest as well.

This goal has attracted a great deal of research in this field, especially research focused on selecting and recommending LO. The most commonly used recommendation techniques are collaborative filtering and data mining. The majority is based on existing information on educational resources (contact-based approach), on the actual user (collaborative approach), or on a combination of both (hybrid approach).

At present there is no platform that can perform this type of search or provide recommendations for LO, while being deployed within a Cloud Computing environment.

3 Proposal: AIREH

The main objective of the present study is to recover LO from LOR by using federated searches. The search system can also filter and classify data according to a set of rules. These rules are generated according to the metadata that describe LO and provide the information required to offer the most appropriate educational resources to each user. The proposed system is presented as an intermediary point of communication between the LORs, the LOs that they store, and the end users.

Given the highly heterogeneous nature of this context, the use of MAS, which takes social norms and organizations into account, was selected to develop the federated search system. This type of architecture makes it possible to describe the functionality using a set of roles, each of which can be carried out by any agent with the necessary capabilities. In practice, this new concept was critical since the changes that can take place within the context can be easily introduced into the system.

Figure 1 displays a structural diagram of AIREH, and presents a high number of important characteristics (and products): UserUnit, SearchUnit, CataloguingUnit, StatisticsUnit, and the Administrative Unit, each of which follows a congregation type pattern within the organization of intelligent agents. The provider and consumer roles are also presented.

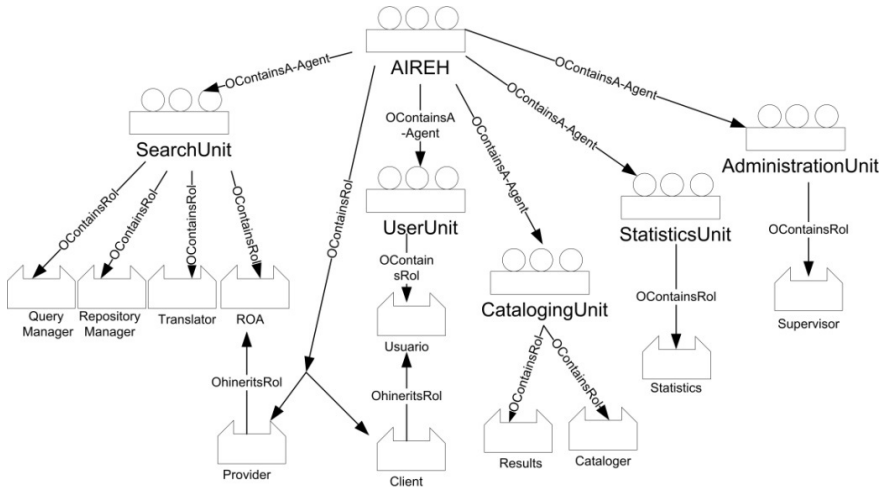


Fig. 1 AIREH functional structure diagram

The complete process comprises three main phases:

1. The selection of the LOR, in which the best repositories are selected according to the statistical parameters that have been gathered from previous queries.
2. Once the LORs have been selected, the next step is to recover the LO according to the search performed by the user. Additionally, it is during this phase that the obtained results are homogenized to facilitate the next phase of the process.
3. The final phase of the process includes a second filtering phase that takes into account the aspects of quality, such as size, completeness, etc., of the metadata. System users also evaluate the objects during this phase, using a voting system in previous searches. The following section provides a more in-depth review of the system.

3.1 Recommendation Strategy

A recommendation system is a tool that predicts user likes according to their characteristics, interests or abilities, based on previously obtained information. There are various techniques based on Artificial Intelligence (AI) which are oriented to carrying out these tasks. One of them is CBR.

The purpose of CBR is to solve new problems by adapting solutions that have been used to solve similar problems in the past [3]. The primary concept when working with CBRs is the concept of case. A case can be defined as a past experience, and is composed of three elements: a problem description which describes the initial problem, a solution which provides the sequence of actions carried out in order to solve the problem, and the final state which describes the state achieved once the solution was applied. A CBR manages cases (past experiences) to solve new problems. The way cases are managed is known as the CBR cycle, and consists of four sequential steps which are recalled every time a problem needs to be solved: retrieve, reuse, revise and retain.

A CBR system depends on the representation of each one of the cases. In this study, the system was designed to offer great strength and flexibility, which makes it possible to adapt the problem to each particular case. Each case is divided into the following main components:

- A set of attributes referred to as *target*, which contains the definition of the problem, that is to say, the query.
- A set of attributes associated to the previous user interactions.

The CBR system is initiated by a new request made by the user to search for LOs. At that moment, the CBR system is executed. The information contained in the new case at the beginning of the execution cycle of the CBR system is defined by the following tuple:

$$c = \{T, u_i, x_i\} \tag{1}$$

Where T refers to the set of attributes defined in the target $T = \{title, language, keywords, format, \dots\}$, u_i is the user identifier, x_i is the value associated with the final solution.

Using the information defined in (1), the reasoning cycle for the CBR system is initiated. Figure 2 illustrates the reasoning cycle. During the retrieve phase the metadata for the learning objects are downloaded from different repositories using simultaneous searches. The information related to the recovered objects is recovered from the cloud. The Slope One method is applied during the reuse phase in order to predict the degree of relevance of the recovered LO. Finally, during the revise and learning phase, information related to the user’s final assessment is stored. The following section explains the different steps for the reasoning cycle in greater detail.

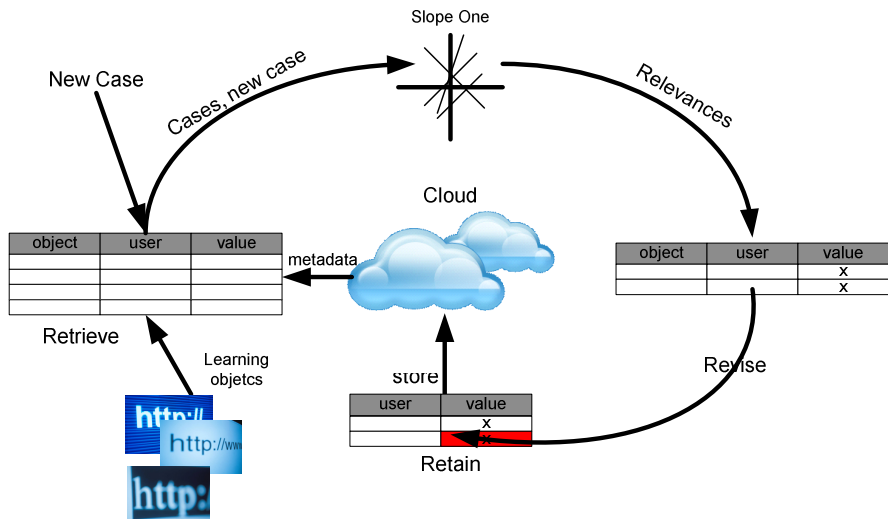


Fig. 2 CBR system implemented in AIREH

3.1.1 Retrieve

During this phase, the LOs are downloaded from the repositories according to the indicated targets. Relative to the LOs, existing information for the stored objects is then retrieved from the cloud. Once the information has been recovered from the repositories and the cloud environment, different cases are obtained according to the structure indicated in (1).

3.1.2 Reuse

The information listed in table 1 is obtained from the data found during the retrieve phase. Each cell contains a value v_{ij} that represents the user's evaluation of the learning object.

Table 1 Information retrieved from the cases.

	LO ₁	LO ₂	LO ₃	...	LO _m
u ₁	v ₁₁	v ₁₂	v ₁₃	...	v _{1m}
u ₂	v ₂₁	--	v ₂₃	...	v _{2m}
u ₃	v ₃₁	v ₃₂	--	...	v _{3m}
...				...	
u _n	v _{n1}	v _{n2}	v _{n3}	...	v _{nm}

The average is calculated for each pair of individuals (2). The final averaged values could be combined according to (3), with a weighted average relative to the number of predictions that exist for each article.

$$\bar{d}_{ij} = \frac{\sum_{k=1}^{m-1} (v_{ik} - v_{jk})}{m-1} \quad (2)$$

Where v_{ik} represents individual i for which the unknown value is being calculated, m is the number of values that exist for both articles i and j (if v_{ik} is unknown, v_{jk} will not be considered in the calculation), v_{jk} is individual j .

$$x_{ik} = \frac{\sum_{j=1}^{n-1} m_j \bar{d}_{ij}}{\sum_{j=1}^{n-1} m_j} \quad (3)$$

Where v_{ik} represents individual i for which the unknown variable for k is calculated, m_j is the number of values that exist for category j , v_{jk} is individual j .

3.1.3 Revise and Retain

During the revise and retain phase, the user rates the objects retrieved during the reuse phase. The values are then stored in the cloud for future retrievals.

4 Experiments and Conclusions

The Merlot¹ and Lornet² repositories were used to carry out test son the LO. These repositories were used to perform searches by 40 users on a group of 60 different key words, taken from the UNESCO code. Each user input a key word and then analyzed the predictions made for the previous 15 predictions. The values were assigned to each item on a scale of 1 to 5. The implementation of the algorithms was based on the Apache Mahout library, which provides techniques such as Map Reduce, allowing a high level of efficiency in multiprocessing systems.

The first step was to compare the execution times for different alternatives to collaborative filtering in order to determine the viability of the different solutions. The execution times were based on simulated data, starting with the first test of 500,000 pieces of data and a second of 5,000,000. Table 2 lists the calculation times to obtain the recommendations.

Table 2 Information retrieved from the cases

Elements	KNN	Slope One	SVD
500.000	43s	39s	38s
5.000.000	6:37s	5:36s	5:52s

In order to analyze the efficiency of the CBR system, the predictions were compared with other methods of collaborative filtering. The techniques selected were KNN (K-Nearest Neighbour) and SVD (Single Value Decomposition).

While the different times for constructing the recommendations are very similar, the difference is due to the fact that the KNN algorithm needs the same execution time for any prediction made for a different user, while the Slope One and SVD have a prediction time for execution of less than one second, regardless of the user. The results shown in Table 3 indicate the average error values obtained by the methods indicated in each column. The weighted values are based on a scale of 1 to 5.

Table 3 Information retrieved from the cases

KNN	Slope One	SVD
1.30	0,76	0.78

¹ MERLOT, Multimedia Educational Resource for Learning and Online Teaching (<http://merlot.org>)

² LORNET, Learning Object Repository Networks (<http://www.lornet.com>).

The results in Table 2 indicate that Slope One provided the best results, although very similar to those obtained by SVD. The reason for not using SVD is that it is necessary to determine statistically the number of elements that reduce the dimensionality, which would involve the analysis of the value with subsequent executions.

The system is still in a process of development and undergoing more detailed testing, which will allow for more extensive results in the future. With AIREH it is possible for the user to retrieve LO efficiently and simply, since it allows the retrieved elements to be filtered according to each user and their previous actions.

Acknowledgments. This work has been partially supported by the MICINN project TIN 2009-13839-C03-03.

References

- [1] Ercan, T.: Effective use of cloud computing in educational institutions. *Procedia Social and Behavioral Sciences* 2, 938–942 (2010)
- [2] Sultan, N.: Cloud computing for education: A new dawn? *International Journal of Information Management* 30(2), 109–116 (2010)
- [3] Kolodner, J.: *Case-Based Reasoning*. Morgan Kaufmann (1993)
- [4] De la Prieta, F., Gil, A.: A Multi-agent System that Searches for Learning Objects in Heterogeneous repositories. In: *Trends and Strategies on Agents and Multiagent Systems: 8th International Conference on Practical Applications of Agents and Multiagent Systems*, pp. 355–362 (2010) ISSN 1615-3871 (Print) 1860-0794 (Online)
- [5] Chiappe, A., Segovia, Y., Rincon, H.Y.: Toward an instructional design model based on learning objects. *Educational Technology Research and Development* 55, 671–681 (2007)
- [6] Dublin Core Metadata Initiative (DCMI), <http://dublincore.org> (accessed 2007)
- [7] IEEE 1484.12.1-2002, Draft Standard for Learning Object Metadata. The Institute of Electrical and Electronics Engineers, Inc.
- [8] Simon, B., Massart, D., Van Assche, F., Ternier, S., Duval, E., Brantner, S., Olmedilla, D., Miklos, Z.: A Simple Query Interface for Interoperable Learning Repositories. In: *1st Workshop on Interoperability of Web-based Educational Systems*, Chiba, Japan (2005)
- [9] Lujara, S.K., Kissaka, M.M., Bhalaluseca, E.P., Trojer, L.: Learning Objects: A new paradigm for e-learning resource development for secondary schools in Tanzania. In: *World Academy of Science, Engineering and Technology*, pp. 102–106 (2007)
- [10] SCORM 2004, 4th edn. (March 2009)
- [11] Ternier, S., Verbert, K., Parra, G., Vandeputte, B., Klerkx, J., Duval, X., Ordóñez, V., Ochoa, X.: The Ariadne Infrastructure for Managing and Storing Metadata. *IEEE Internet Computing* 13(4), 18–25 (2009)
- [12] Wiley, D.A.: Connecting learning objects to instructional design theory: A definition a metaphor, and a taxonomy. In: Wiley, D.A. (ed.) *The Instructional Use of Learning Objects*, IN, Association for Educational Communications and Technology, Bloomington (2001), <http://www.reusability.org/read/> (retrieved March 23, 2009, from the World Wide Web)

- [13] Polsani, P.: Use and Abuse of Reusable Learning Objects. *Journal of Digital Information* 3(4), arto 164 (2003)
- [14] Hatala, M., Richards, G., Eap, T., Willms, J.: The Interoperability of Learning Object Repositories and Services: Standards, Implementations and Lessons Learned. In: 13th World Wide Web Conference, Educational Track, New York, pp. 19–27 (May 2004)
- [15] European Committee for Standardization, A Simple Query Interface Specification for Learning Repositories (November 2005)
- [16] Lagoze, C., Van De Sompel, H., Nelson, M., Warner, S.: The Open Archives Initiative Protocol for Metadata Harvesting. Open Archive Initiative. Version 2.2 (2002)
- [17] De la Prieta, F., Gil, A.B.: A Multi-agent System that Searches for Learning Objects in Heterogeneous Repositories. In: Demazeau, Y., Dignum, F., Corchado, J.M., Bajo, J., Corchuelo, R., Corchado, E., Fernández-Riverola, F., Julián, V.J., Pawlewski, P., Campbell, A. (eds.) *Trends in PAAMS*. AISC, vol. 71, pp. 355–362. Springer, Heidelberg (2010)
- [18] Gil, A.B., De la Prieta, F., Rodríguez, S.: Automatic Learning Object Extraction and Classification in Heterogeneous Environments. In: Pérez, J.B., Corchado, J.M., Moreno, M.N., Julián, V., Mathieu, P., Canada-Bago, J., Ortega, A., Caballero, A.F. (eds.) *Highlights in Practical Applications of Agents and Multiagent Systems*. AISC, vol. 89, pp. 109–116. Springer, Heidelberg (2011)