A service-oriented architecture for real time data collection in e-learning context: SCORM runtime environment case study.

Youssef Ait ouguengay, Nour-eddine El faddouli, Mohamed Khalidi Idrissi, Samir Bennani

RIME Team, Departement of computing
Mohammed Vth University- Agdal, Mohmmadia School of Engineers (EMI), Rabat, Morocco
ouguengay@gmail.com, faddouli@emi.ac.ma, khalidi@emi.ac.ma, sbennani@emi.ac.ma, http://www.rime.emi.ac.ma

Abstract: - Digital tracks represent the information about activities of users during their interactions with a computing platform. In e-learning environments, the analysis of these tracks is considered as an information source of obvious importance: the Effectiveness of the monitoring process of learning depends on it as taking good decisions like the update of the available learning resources and the personalization of the learning path of learners according to their activities and their perpetual changing profiles. In this paper we propose a web service based system for tracks collection. A Service oriented layer for data collecting, based on the standardized CMI data model is introduced. So as to construct a complete data model matching the needs of learning tracking, we suggest defining an extension for the CMI data model adopted by SCORM to take into consideration the activities of communication between different learning actors within the learning management system. Components charged of data preprocessing and transformation are also, but briefly, described.

Key-Words: data collection, tracks analysis, SOA, e-learning, SCORM, data model, interactions.

1 Introduction
Learning management systems (LMS) are widely used in distance learning programs. An LMS is a software solution whose the main functions are to allow tutors to manage their process of learning as to deliver courses to students, preparing platforms for testing, starting discussions, producing course content, manage the distance learning scenario, etc. These learning environments manage communications between delivered resources and customers through an API (Application Program Interface). Pedagogical objects and learning activities (courses, resources, etc) are served by the API, and users’ interactions with the learning content can be enhanced.

By definition, the various objects and interactions maintained by the LMS should be fully described by the metadata handled by the LMS, the corresponding information is a structured data stored somewhere in the system for the sake of management and analysis. Particularly, in the field of standardization of distance learning, IEEE 1484.11.1 [1] defines a data model dedicated to describing the metadata of learning objects and interactions. The IEEE data model is implemented by the most of the current specifications and standards on e-learning field (i.e, SCORM, IMS-LD, QTI). It is a rich source for learning and interactions tracking and can refer to any form of computer based learning.

We intend in this paper, to give an overview of the proposed system and particularly describing the process of interaction’s information collection taking as a core the IEEE 1484.11.1 model. In the next section we give a state of the art of the interactions analysis that we can find in existing e-learning literature. We describe, in the third Section, the overall architecture of the proposed system for traces collection, particularly the intermediate layer designed for collecting the raw traces based on the standard model of communication IEEE 1484.11.1.

2 Tracks analysis In distance learning environments
Different approaches to trace analysis of interactions in the learning process tend to have specific
objectives related mainly to the principle actors of learning: learner(s), tutor and pedagogical content. According to [2], these objectives can be classified into three main categories: Optimizing the quality of educational resources available on the system and the learning scenarios which are designed by teachers, improving methodologies of making knowledge to learners, and finally, designing tools for tutors and external observers (researchers, etc) for supporting learners. The tutor, being the cornerstone in the learning process, uses these tools to assess, in one hand, the performance of learners in their learning process and come to their aid when necessary, and in the second hand he assess the quality of teaching resources which are used.

2.1 E-learning tracking systems: A state of the art

Interactions between the system of learning and users generate traces of user data and activities. They can be observed and recorded, mostly in log files of the LMS or in a database. Number of pages visited in a course, the results of evaluation tests, details of the choices and training courses followed, time spent per module, etc. are examples of information directly related to learning activities. We also distinguish the traces of the activities of communication between users of the system: messages exchanged between tutors, administrators and students during participation in discussion forums, etc.

To achieve their ends, the first task of user interactions analysis systems in a LMS is the collection of raw traces data of interactions. According to the objectives laid, these traces can be qualitative, related to concepts such as cognitive collected through a survey or a test. it can be also quantitative, collected directly from the various sources of data system such logfiles or the LMS database. Data (binary or text) not being all directly usable, a second role is required, that of transforming these traces to an understandable, readable and useful information through a method of analysis of data to be viewed or to measure a predefined indicators.

As the volume of work produced in this sense is vast, it is not possible to provide a complete exam of all works achieved on systems of trace analysis. However, we cite in the following some referenced works.

An approach directed by models is proposed in [3], based on the WEBM architecture associated to the CIM model (Content Information Model) for the management of traces. This approach permits a fast Integration of new components, but considering the nature of the CIM model that has a canonical format of objects, an initial mastery of the whole semantics of the domain is necessary in unified way (that is in this case the e-learning field), which proposes an important maintenance effort and significant updates in case of need.

A language specific UTL (Usage Tracking Language) is defined in [4]. It permits to describe according to a specific model the traces independently from the e-learning operation allowing a structured exploitation of traces.

[5] proposes a formal model of a trace-based system which defines several levels of treatment from collecting, processing till the semantization of log files data present on the e-learning platform. More recently, [6] was based on the system proposed by [5] and presents an application to restructure the raw data from data archives (Moodle LMS) to facilitate the use of their footsteps and help data semantization by ontologies. Also in more recent works, [7] build a system for trace analysis for a collaborative learning environment.

Other propositions have focused on standards of learning: [8] proposes a boosting of the LMS API so it can support the advanced features specially tracking. In the same way [9] have proposed extensions for the runtime data model in order to integrate communication activities via email within the SCORM based learning objects, while the mail communications are usually deployed outside of the learning process.

The work around trace-analysis systems of e-learning environments aims to implement a specific data mining solution in order to resolve specific problems and the technologic deployed solutions are often based on software agents whose connections are point to point, which increases the number of connections required to implement solutions and rises then multiple update and maintenance issues. On the other hand, log files are the main sources of data used in such systems, they are certainly cheap sources of information and can be technically implemented easily, but instead the log data processing is an expensive operation [10].

2.2 Principal Advantages of SOA

Service oriented architecture is an approach of architecture where the global logic is decomposed into distinct units of logic. SOA architectural platforms present many advantages:

- Integration and interoperability: Adopting a web SOA based system enables integration of functionality in a modular fashion and
gives a loosely coupled system.

- Intrinsic reuse: A more complex system can be designed from the restructuring of existing services that has standardized XML based interfaces and make data flow models reusable. This flexibility allows the easy definition of new systems by declaration, which are immediately usable and facilitate the maintain of services.
- Benefits in investment: the cost of integrating contemporary components into an existing system is noticeably decreased since the Web service technology has resulted in a large market of adaptive solutions.

3 A conceptual architecture for learning tracking: SCORM runtime environment case study

In order to establish a tool for data analysis independent of platforms and languages in which the learning system components would be based, we envisage a web service oriented architecture. The system as proposed permits learning interactions analysis. It’s basically decomposed on three processes: raw data collection, transformation of data collected in a specific format to prepare it for analysis and the analysis phase itself, according to indicators that are designed to help the system actors in their tasks and goals (tutor, learner, course designer, researcher, etc). Finally, a results display format is considered.

3.1 The CMI data model standard:

The adoption of learning management systems arouses an increasing interest among decision-makers to the concerns of standardization, interoperability and reuse of learning resources. These problems have set up several research areas of global standardization bodies. Among these efforts we find the CMI (Computer Managed Instruction) data model for communication between content and learning system. In this context, and to describe the interactions of users in a learning environment, we propose a reuse of this data model in the tracking system.

The CMI is a standardized data model that describes all data exchanged between the learning resources and the environment LMS via an API. These data cover information on learners, learners' interactions with learning objects, the status of activities (successes, completeness,...), etc. It also trace the performance of learners in relation to learning objectives. The following information architectured standard CMI allow easier analysis of traces of learners.

Historically, CMI was originally proposed by the AICC (Aviation Industry Computer-Based Training Committee) and firstly suited to the SCORM specification. However, this standard provides an 'abstract' data model, and thus independent of any technology implementation type, which describes the interchangeable data between a learning object and a learning management system through a runtime environment (RTE). Since 2004, CMI has been adopted by IEEE as a full standard titled: "1484.11.1: Data Model for Content to Learning Management System Communication".

![Fig. 1: Conceptual diagram of a learning environment](image)

The traditional use of the CMI model is during the execution of the learning object [11]: the Runtime Environment (RTE) that manages information about user interactions with the learning object (Figure 1) uses the CMI for the validation of the generated data streams. The major benefit of the model is that it can be implanted as well to describe the communication between client and server side of the support services (RTE), or simply archive the information generated in a semantized standard format for analysis and reporting needs. In this sense, the proposal we make is to reuse the standard for the purpose of collecting data. This collection, itself, is part of the broader objective to trace and analyze user interactions with the learning environment.

In an environment conformed to the CMI requirements, each activity served on a learning platform can be composed of one or more learning objects (LO). A learning object may be part of a course or courses taught throughout the learning process. At the request of the learner, the execution of each LO occurs between the sending of the call of two procedures: the initialization and the suspension of the activity.
During the execution of the LO, the runtime environment is supposed to manage the data model exchanged between the client and the LMS. At the time of suspension of an activity relative to a user, a course and a specific session, data is available through the LMS. In the division of responsibilities between the LMS and the LO, defined by the SCORM specifications [8], it imposes no constraint that the LMS keeps the RTE data after the suspension of the learning session. In reality, few LMSs, at least based on SCORM, are offering support for the extraction and reporting of qualitative information interaction during the execution of the LO [12].

Among the roles assigned to our proposal, explained in the next section, is precisely to store the metadata from the executed LO instance in the RTE to a learning record store (LRS). The approach is meant to be more general, as it permits to mule information produced by external services.

3.2 Overall system architecture for tracks analysis

The first component of the proposed architecture is the middleware responsible of collecting (extracting) data from the learning system runtime or from other data sources (web logs, surveys, questionnaires, etc.) That may be available.

The collection task is the only component of the analysis system that interacts with the learning environment. The traces collected may include any data of interest (learning activities, communication, etc). The flow of data between the LMS and the collection service is enabled by an extended data model which includes elements of learning activities described by the CMI model and augmented with elements describing other external activities of learners communication that are not included in the CMI standard.

The basic component introduces a virtualization layer that intermediate between the LMS supporting the learning activities and the tooling system designed for tutors and observers. This layer consists of a data bus whose main task is the preprocessing of data exchanged between services and applications and their aggregation traces usable by the analysis service.

The main functions of this layer of virtualization are:
- Interception and filtering of messages in real time and querying further data sources in case of need (DBMS log files, etc.).
- Merging data into a common data model suitable for trace analysis needs.
- Routing and saving the results of analysis for viewing or archiving.

Figure 2 shows the synthetic scheme of the tracking system. From the perspective of e-learning trace analysis, basic information gathered by the component of collection are determined by pre-defined indicators and objectives of analysis already established [14]. This simple SOA based architecture enables rapid implementation and reuse of services. The flow of data between the LMS and the service bus is validated against a common data model, integrating the learning activities and communication subjects’ analysis (see next section).

Fig. 2: architecture of the proposed e-learning traces analysis system

All data of interest generated cross the learning process, is collected and routed either to the analysis service or to the Learning Resources Store database so that it can be available for future analysis.

The analysis process passes through a stage of transformation of data: It comes to verification of all data prior to analysis (including pre-processing steps, eliminating irrelevant data and empty messages, etc.). The data transformation is highly dependent on the adopted approach and analysis methods (statistics, artificial intelligence,...). The operation of the analysis itself is meant to measure the pre-defined indicators and draw interpretations with the ability to make recommendations for decision-making on the basis of indicators calculation and deliver the data of these results to the service of visualizations.

4 Conclusion:

The collection of data is an essential step of the interactions analysis; it is an intrinsic task to the core of the system. Since, in SOA architectures, the messaging patterns are handled between services, exploring the data bus for collecting data in real time is our first step to a system which concerns the eventual needs to real time learning interventions.
The proposed method for Analyzing e-learning Traces in this paper, is based on a web service oriented architecture. Particularly, the intermediate layer proposed, charged of collecting evidences, is based on the CMI standard data model. An extension to this data model including the external elements of learning was described. We propose the defined extension in the consideration of other communication services used in the learning platform and which can be considered part of the learning process especially in the case of collaborative learning approaches. A prototype platform which details each component of our system is needed. In the prospects for our proposal, we look to validate the enriched CMI data model scheme and to identify the possible scenarios for using it in a real word implementation.

REFERENCES: