



**HAL**  
open science

# Educational Webportals Augmented by Mobile Devices with iFrimousse Architecture

Florent Carlier, Valérie Renault

► **To cite this version:**

Florent Carlier, Valérie Renault. Educational Webportals Augmented by Mobile Devices with iFrimousse Architecture. 10 TH IEEE International conference on advanced learning technologies, Jul 2010, Sousse, Tunisia. pp.236-240, 10.1109/ICALT.2010.72 . hal-01226436

**HAL Id: hal-01226436**

**<https://hal.science/hal-01226436>**

Submitted on 30 Apr 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Educational webportals augmented by mobile devices with iFrimousse architecture

Florent Carlier and Valérie Renault

*Laboratoire d'Informatique de l'Université du Maine*

*Le Mans, France*

*Email: {florent.carlier, valerie.renault}@univ-lemans.fr*

**Abstract**—In the context of educational devices, and of Web portals in particular, the learner and the tutor can be confused by the sheer volume of information available, the evolution of the educational practices in relation to mobile supports and the problems of asynchronous interactions. Our task is to find a way to augment the Web's educational portals using animated characters and mobile supports with a view to relocating synthetic information adapted to the active user (learner or tutor) in a particular context. We present an epiphyte architecture which works with an existing educational portal. We illustrate this work using a scenario in which a mobile system, such as an iPhone, allows the teacher to see the progress of his students, in real time, during practical sessions.

**Keywords**—User assistant, mobile supports, epiphyte architecture, Web portals augmented

## I. INTRODUCTION

With the explosion of information technology, communication and on-line publishing, e-learning portals, courses and resources have been widely developed. The objective of these on-line environments is to facilitate distance learning, to announce changes in teaching practices, and to interact with the traditional classroom. Nowadays, a course designer has a wide range of tools at his disposal, allowing him to - for example: give resources to students and share the educational material, organize on-line discussions, gather and learn from others' learning situations, follow and help remote students.

A large number of learning portals are based on Content Management Software (CMS), such as Moodle [11], Claroline [5] to quote the most common. CMS allows course designers to structure the information at their disposal without the need for complex computing knowledge. However, numerous authors, such as Jafari [10], underline the limits of e-learning portals for the teachers as well as the student. From the point of view of the student, a learner is alone in front of the computer (no direct communication with the teacher), thus collaboration between the learners themselves is often ignored and educational methods are still not adapted to the needs of each. Besides this, the quantity of information supplied by these e-portals is so vast that both learners and teachers get lost amongst the wealth of courses available, exercises to do, news, forums etc.

Add to this the fact that, apart from these portals, students now have more and more access to a complete

workspace can include, for example, their schedule and personal workspace, the home pages of their university campus, their library and their teachers, thus it is easy to understand how a student can be quickly lost within the mass of information available to him. One of the consequences of this is that he can miss important information completely. Furthermore, a learner, during the day, does not always have access to the same media technology all the time, he can use his personal computer, a computer room for practical work or an information terminal and other media in a library. There may also be situations where he needs media support but none is available : such as consulting his schedule whilst in the hallway (he may not always be near a noticeboard, for example), having discussions with other students on public transport about an exercise he has to do etc.

It is the same for tutors who put courses or tasks (as defined in Moodle) on-line. They too may be overwhelmed by the amount of information available to help them evaluate students. In contrast, most users (whether students or tutors) have mobile devices, which are becoming more and more powerful, "embedded" in their mobile phones. With the merger of PDAs and smartphones and the expansion of WiFi, we can now consider many applications to be ubiquitous, and adaptable to the work situation of the user according to context.

Our problem is to increase (as defined in the augmented reality) educational Web portals through animated agents and mobile carriers to disseminate the summarized information tailored to the active user (learner or tutor) in a specific context.

iFrimousse is an epiphyte architecture easily configurable to increase the use of e-learning portal, using embedded systems and embodied conversational agents (ECA). We chose the term 'epiphyte', in parallel with the notion of epiphytes plants (such as orchids), because our application overlaps on an already existing system without modifying it. In this paper, we focus on the axis of the mobile media, even if the iFrimousse architecture is developed with both constraints of mobility and animated agents. We let the reader refer in [3] for more information about the agent aspect of our platform. We present the epiphyte architecture developed for an agent, and the add-on used to allow this architecture to support mobile applications. Finally, we show how a mobile support can be used to synthesize the relevant

information for the user in a given context, in addition to the web portal which it handles.

## II. BACKGROUND RESEARCH: E-LEARNING PORTALS INCREASED WITH MOVING SUPPORTS

### A. *Relocation of the online help on moving supports.*

Portals like Moodle or Claroline allows teachers to prepare for practical sessions or to upload homework between sessions. In a typical scenario for using the platform, each student (or pair of students) has a task to download a set of exercises to do during the session, which he must then post, so that the results can be measured.

During a course session - depending upon the degree of autonomy given to each student - a teacher may answer individual questions "in the background", whilst not necessarily taking the time to analyze each response for the benefit of all. More often, teachers will favor students who ask questions, as opposed to those who have problems. Our proposal in this case is to provide the teacher with a summary of the progress of each student on a mobile platform which he can keep with him whilst he moves around the classroom. The criteria for acceptability of this summary are: in a first time, a visual representation of the progress of all students in order to quickly distinguish those who are not the same level from the others; in a second time, a speedy access to information on the progress of individual students.

### B. *Client-server approach*

Usually, three types of approaches can be admitted in order to set up architectures for web applications [12] [14]: Server oriented approach (SOA) [2], Client oriented approach (COA) and Proxy oriented approach (POA) [8]. In our case, COA and POA approaches have the disadvantage of requiring the user intervention to configure them. We refer the reader to the bibliography for more informations about COA and POA approaches.

The SOAs are approaches where treatments from a given architecture are relegated to the hosting server. Typically, these approaches are totally transparent for the user: he doesn't even know that the content he receives has been intercepted and modified. They allow filtering of the whole data stream, and their main interest is that a server will be able to produce much information about the architecture, thus facilitating its conception and development. Moreover, the control on what is done is absolute: the client's machine is never responsible for any treatment. On top of that, linking with a connate system do not require changes or design anticipation.

However, such an architecture must be able to manage simultaneous access to it, and it is imperative to ensure that the changes operated on the server are minimal, because the architecture should not severely impact the performance of a whole production server.

## III. IFRIMOUSSE: ARCHITECTURE EPIPHYTE EDUCATIONAL WEB PORTALS

Our main objective is to define an architecture sufficiently generic and easily configurable as to enable it to increase the use of e-learning portals, using existing embedded systems. This assistance will, according to context, be located on the user's machine via an embodied conversational agent (ECA) or on a mobile support. As the site has not necessarily been designed to develop new remote services, we propose a method which shows the location of the relevant content (relevant for the purposes of the designer of the platform) within a page of an existing website. This information will then be able to link the website to a mobile device, synthesizing other information in conjunction with the site.

Figure 1 shows the proposed architecture and explains its mechanisms. Our context is a bit particular, we wish to develop an architecture generic enough to create a link between systems using ambient to existing learning platforms. We especially want to link mobile systems (with or without ECA) with existing portals without deeply modifying them.

Web platforms have a set of HTML (Hypertext Markup Language) which does not disclose the locations of semantic information useful for our needs. Our first task was therefore to define an architecture for locating from within the site, information related to the course (author duty, deadlines, etc.). We offer a comprehensively layered server oriented architecture, based on the concept of middleware.

We propose a layered architecture, assembled from four parts: 'network', 'tag location', 'mobile system' and 'dynamic ECA'. Each one of them uses a specific approach, and is designed with purposes of reusability and genericity. More important, we want our architecture to be totally transparent for the user.

The first one is the 'network' layer, and can be seen as the architecture's groundwork. The system's efficiency will rely on its ability to intercept all the needed data streams. The SOA approach, based on the middleware concept, is basically a communication layer that allows applications to interact across various software and network environments. This concept has been standardized for the Python language, under the name WSGI (Web Server Gateway Interface) [6] and is a specification for an interface between web servers and Python web applications, to promote web application portability across a variety of platforms. The Apache's modular architecture ensures the link between the original web server and the layer, by redirecting the interesting requests to it. The layer consists in a dedicated WSGI server, which is the only entry point for the whole architecture, developed using a Python framework [9]. This design allows us to only catch the desired traffic, and to let all the unneeded content through the standard path. In this way, the web server performances for the target platform are not impacted much, and the other web applications running on it aren't impacted

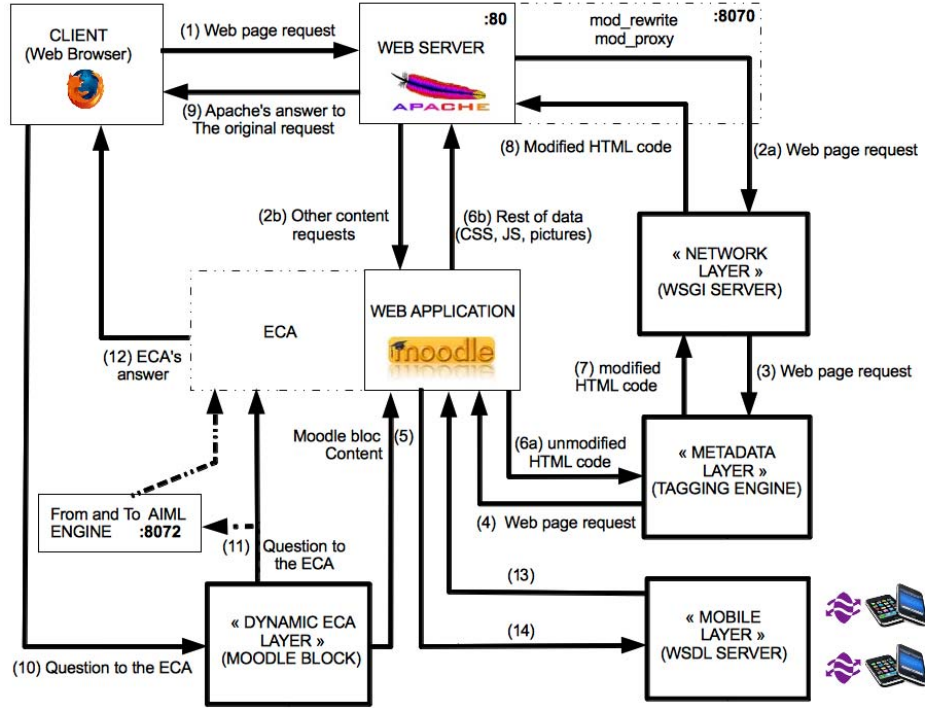


Figure 1. Organization of the epiphyte architecture

at all. Moreover, the user doesn't even notice that he is redirected to another server.

The second layer is the 'tag location' layer. Its mechanism consists in inserting additional content in the pages transmitted to the client browser. In the present case, this additional data will be intended for localization means, under the shape of tags considered as anchor points for the last layer. This layer is structured with a tagging engine, responsible for tag insertions, and a system of modules, each one dedicated to a specific platform. Each tag has a type (useful in group analysis, for example) and an id, which is unique on a given page. The tagging decision is based on pattern recognition inside the learning platform pages code. Another feature proposed by this layer is the generation of tag lists of a page in the XML format, allowing a third-party system to question the architecture, in order to map the platform, for example. The development has been assured using a Python HTML/XML framework [1]. All this tagging work ensures that we get usage logs from the architecture, and that the logs database will remain valid, since our two first layers can't be bypassed by the user, unlike the POAs.

The third layer is the 'mobile system' layer. It aims to create a bridge between the e-learning platform and the range of mobile systems available. In the context of object-oriented architecture, we use Web services, enabling

interoperability between different software systems (agents) on a computer network [4]. More specifically, we use the W3C standards, the system interface is defined by a language readable by the computer client. The WSDL (Web Services Definition Language) [16] language describes the interface to the service. By using XML Schema, WSDL defines the input parameters for making and returning calls to the Web service. Other software systems communicate with the Web service according to their description using SOAP (Simple Object Access Protocol) [13]. SOAP transports materials and other serialized XML data and the call of remote process. In order to be able to use a 'Web services' and to program a client (in our case, a mobile system such as a smartphone), it is necessary to understand the definition. A WSDL file has been identified as a link with the Moodle server. In this case, the use of our architecture with another type of CMS, the designer will create a WSDL file adapted in order to maintain interoperability between the server and the mobile system.

The last layer is the "dynamic ECA" one. This last layer manages all the interactions with the client, particularly the interactions between the ECA and the user. We develop a specific API offering base features for moving an entity such as a ECA. The parameters used are the desired displayed text, the used angle of the ECA for pointing content, the



Figure 2. iFrimousse : Mobile application on iPhone/iPod Touch of Apple

return of the ECA after moving, the movement of the dialogue box and the attitude of the ECA. We invite the reader to refer to [3] for more information about this layer.

Figure 1 summarizes this architecture. The client's browser sends a request for a web page to the web server (1). The request is then redirected to the "network" layer (2a), while all the requests for uninteresting data (pictures, script files, style sheets) are directly sent to the web application (2b). This layer then gives the request to the "metadata" layer (3). Once the request has been transmitted to the application (4), the "dynamic ECA" layer sends its data (5) for being integrated in the resulting unmodified HTML code, which is given back to the "metadata layer" (6a). Meanwhile, the uninteresting data is directly returned to the web server (6b). Then, the tagging engine of the "metadata" layer modifies the HTML code, and sends it to the "network" layer (7), who the last role is to redistribute the page to the web server (8), which will answer to the client's original request with the prepared page (9). With this architecture, the client can ask a question to the ECA via the "dynamic ECA" layer (10). The layer will decide what to do with the question (treat it directly or pass it to the ECA (11)), then the client will receive the answer to it (12). If the user is in context of mobility (13), specific services (14) are sent to the mobile layer by the WSDL server. If the user is in a context of mobility (13), specific services (14) are sent to the mobile layer by the WSDL server.

The performance of the whole system hardly impacted at all: the conducted benchmarks show that the extra treatment time per request is really minimal, and does not exceed 30

milliseconds.

Our architecture was implemented using the Moodle platform and the mobile device iPhone / iPod Touch. The mobile systems used are part of the smartphone category. To test the feasibility of our architecture, we implemented the scenario presented in section II.A to relocate the online help on a iPhone. The simple point of this scenario is to enable the implementation of our architecture in a controlled environment.

#### IV. WHEN THE MOBILE PHONE BECOMES THE ASSISTANT TUTOR

The goal of this application is then to facilitate the teacher's vision of the progress and achievement of his students in real time during practical sessions, and not retrospectively.

The teacher goes through the rows of students with his iPhone to help students complete their practical work. When he approaches an individual student, he can select the icon for that student on his iPhone and study a table summarizing his progress, and will know if the exercises have been completed or not, and thus can give immediate help to the student if needed. For each individual student, the teacher has the following information : a list of exercises on the platform, a progress bar for each exercise, indicating whether a file has or has not been filed on the platform; and a small bar if the file size below a certain pre-defined threshold.

For rapid access to a student, the teacher can select from a list shown in the form of a carousel (Figure 2), the background color of the icon corresponding to the data

obtained from the progress bars for each student. This display allows the teacher to immediately see the overall progress of the practical session and identify students with difficulties.

This representation comes from work on the DividingGuest platform (Girard and al., 2006), an ILE coupled with an evaluation software platform for the study of the impact on the use of an interface with animated characters as an aid to learning. The DividingQuest is an intelligent learning environment type Vygotskian (Vygotsky, 1978) used in English primary schools.

We have implemented this representation by changing the background color of the card gallery of the carousel. The colors red, amber and green represent the 3 levels of progression that the teacher has defined when configuring the epiphytic architecture to complete his tasks on the e-learning platform. The color blue for a student represents neutrality in monitoring current activity, for example, for a student does not wish to take part or has already completed the activity at home. We are in the process of validating our experimental process for the tutor. The next step will be to test this same interface for learners. As in the DividingQuest, students should then be able to benefit from the result of his own learning process throughout the length of the practical session.

## V. CONCLUSION

In order to promote interaction in the learning process, the iFrimousse project aims to achieve a system of mobile and ambient learning which enables a virtual assistant to monitor the learner according to his needs and his work environment by augmenting the portals of existing e-learning.

This article shows the first stage of this project which has led to an architecture epiphyte, that is to say an architecture which can be grafted onto existing portals without modifying the original platform. To achieve this first step, it was necessary to develop locator beacons from knowledge specific to the scope of each site. These tags allow us to locate specific information. We then defined two scenarios, one oriented to the agent, the other to the mobile terminal, allowing us to test our architecture in smaller and more controllable contexts.

Our goal is now towards more complex scenarios which extend the mobility of the active users. Many avenues of research lie open: first, obviously, is to allow the movement of the agent's web portal to support mobile devices, depending on the context of their use. Many questions arise: for example, what happens when a course is finished and the teacher still has an 'alert' on his iPhone? How can the student have access to an interface similar to that of the teacher in order to understand his status in relation to the rest of the group? In what contexts should the ECA move on the mobile terminal in order to help the learner?

As our architecture is epiphyte, different opportunities exist to adapt this platform to other contexts, especially in

the context of home help for a person with reduced mobility. He can combine our architecture with an Automated Web Portal. Our objective is then to allow the person to have an intelligent mobile support, moving with them into his or her home to assist with their daily life.

## REFERENCES

- [1] ANGUENOT J., Python et xml. Proc. of RMLL 2005. (2005)
- [2] BRUSILOVSKY P., DEBRA P., SANTIC T. A flexible layout model for a web-based adaptive hypermedia architecture. Proc. of AH 2003: Workshop on Adaptive Hypermedia and Adaptive Web Based Systems, pages 7786. (2003)
- [3] CARLIER F., RENAULT V. An Epiphyte Architecture to enrich content in an existing educational website., In: 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 8-12 dcembre 2008, Sydney (Australia), p. 135-138. (2008)
- [4] CHAUVET J.-M. Services web avec SOAP, WSDL, UDDI, ebXML..., Editions Eyrolles. (2002)
- [5] CLAROLINE. <http://www.claroline.net/>. (2009)
- [6] EBY P. J. PEP 333 Python Web Server Gateway Interface v1.0. <http://www.python.org/dev/peps/pep-0333>. (2006)
- [7] GIRARD S., JOHNSON H. Design affective educational software products: Sormo, a new method for capturing emotional states . Journal of Engineering Design, Special Issue on Design and Emotion. 1466-1837, Volume 20, Issue 5, Pages 493 510. (2009)
- [8] GIROUX S., PAQUETTE G., PACHET F., GIRARD J. Epitalk, a platform for epiphyte advisor systems dedicated to both individual and collaborative learning. ITS96, Springer-Verlag Lecture Notes in Computer Science. (1996)
- [9] HELLEGOUARCH S. CherryPy Essentials. Packt Publishing. (2007)
- [10] JAFARI A. Conceptualizing intelligent agents for teaching and learning. EDUCAUSE QUARTERLY, 25(3). (2002)
- [11] RICE W. H. Moodle: E-Learning Course Development. Packt Publishing. (2006)
- [12] RICHARD B. Une approche epiphyte pour la conception de systmes conseillers. Thse de l'Universit du Maine. (2008)
- [13] SOAP. SOAP Version 1.2 Part 1: Messaging Framework (Second Edition). <http://www.w3.org/TR/soap12/> (2007)
- [14] SRIVASTAVA J., COOLEY J., DESHPANDE M., TAN P.-N. Web using mining: Discovery and applications of usage patterns from web data. SIGKDD Explorations, 1(2):12-23. (1999)
- [15] VYGOTSKY L.S. Mind in society: the development of higher psychological processes. Cambridge, MA: Harvard University Press. (1978)
- [16] WSDL Web Services Description Language (WSDL) 1.1. <http://www.w3.org/TR/wsdl> (2001)