Fractal Adaptive Web Service for Mobile Learning

Ichraf Tirelli1,2, Mona Laroussi1,3, Alain Derycke1, Henda BenGHezala2
1Université des Sciences et Technologies de Lille, France
2RIADI ENSI campus de la Manouba, Tunisia
3INSAT Centre Urbain Tunis Nord, Tunis, Tunisia

Abstract—this paper describes our proposition for adaptive web services which is based on configurable, re-usable adaptive/personalized services. To realize our ideas, we have developed an approach for designing, implementing and maintaining personal service. This approach enables the user to accomplish an activity with a set of services answering to his preferences, his profiles and to a personalized context. In this paper, we describe the principle of our approach that we call fractal adaptation approach, and we discuss the implementation of personalization services in the context of mobile and collaborative scenario of learning. We have realized a platform in this context -a platform for mobile and collaborative learning- based on fractal adaptable web services. The platform is tested with a population of students and tutors, in order to release the gaps and the advantages of the approach suggested.

Index Terms—Web services, adaptation, fractal adaptation, composition, discovery, orchestration, mobile learning

I. INTRODUCTION

The Web has become an integral part not only of our business, but also of our private life. Complex tasks like arranging travels, booking flights or online banking can easily be carried out over the Internet. Whereas these services are up to now typically accessed through the use of personal computers, the recent development of powerful mobile devices, suitable protocols and improved bandwidths promise to take Internet services a step further.

The vision of a mobile Web in which the computing environment will be composed of various devices that are carried by different users as they go through their daily routine might soon become a reality. In this context, Web Services are recognized as important building blocks of the future Web. [Wagne, 2002]

An effective use of these services can only be achieved through adequate adaptation and proactive service behavior.

Whereas technical issues for mobile web services are thoroughly investigated, work on service adaptation is more and more on progress.

In this paper we present an approach of web service adaptation in the case of mobile and collaborative scenario of learning.

We begin this paper with a list of useful definitions, followed by a summarized view of the research issues in web services adaptation, then we detail our adaptation’s approach, we conclude with a validation, results and conclusions.

II. DEFINITIONS

We are interested in this work to three types of web services (as shown in figure 1): classical web services, mobile web services and mobile adaptive web services.

![Figure 1. Our context and Fields of interest](image)

As shown in figure 1 we are interested to three classes of web services. For these classes we discuss the problem of adaptation.

We treat web service’s adaptation in two different ways. The first is the adaptation of an elementary web service: what and how to adapt for each class of web services. The second one is the adaptation of a composite web service: how to treat the composition of web services in order to obtain an adaptive composite web service by composing a set of adaptive web services, and how to treat the change of adaptation’s parameters for a composite service. We call this way to treat adaptation: fractal adaptation. [Laroussi, 2005]. This concept will be more detailed in the paragraph 4 of the paper.

In the following a list of useful definitions - essentially classical, mobile and adaptive web service- before detailing our approach of fractal adaptation.
Fractal adaptive Web service for mobile learning

A. Web service
A Web service is a software component ensuring interoperability between applications, by minimizing the needs for common knowledge between these cooperating entities.

IBM gives in a tutorial\(^1\) the following definition of the web services: «the web services are the new wave of the web applications. It is applications modular, self-contained and self-descriptive that can be published, located and called upon since the web. The Web services carry out actions going of simple requests to complex processes trades. Once a Web service is deployed, other applications (including Web services) can discover it and call upon it.»

So, a web service achieves a precise task, has a set of inputs and outputs and checks a set of conditions. It has a syntactic and a semantic description.

A web service is described by WSDL language (Web Service Description Language). The calls to the interface are carried out by the means of SOAP (Simple Object Access Protocol). The third fundamental component is UDDI (Universal Description, Discovery and Integration): it is a divided directory on the web, in which each one can publish the web service that it wishes to make divided.

B. Mobile Web Service
A mobile service [Fuggeta, 1998] is a software component which can move of a site to another during his execution to approach data or resources. It moves with its code and its data, but also with its state of execution. The service decides itself in an autonomous way of its movements. Mobility does not replace the capacities of communication of the services but supplements them (the distant communication, less expensive in certain cases...). In order to satisfy the constraints of the networks of big size or without wire (latency, not permanence of the bonds of communication), the services communicate by asynchronous messages.

We take in the continuation some examples of the mobile services in addition to the description of such a service.

C. Mobile Adaptive Web Service
Several definitions of adaptable web service exist; we retain definition given in [O’Looney, 2001]:

“Personalization and customization refers to the ability of an Internet Web site or service to be shaped or reshaped so as to better meet the individual needs or wants of a user."

So, a Web service is known as adaptable if some of its internal, operational mechanisms (sending of messages, displacement...) or functional (behavior) can change in the course of execution or on the fly according to devices,

\(^{1}\) http://webservices.xml.com/pub/a/2001/04/04/webservice s/index.html

context, user preferences, ... The service controls itself its own evolutions.

An adaptive mobile web service is a mobile and adaptive service.

D. Fractal Adaptive Web service
We define a fractal adaptive web service as a service which is adaptive in his description -definition- and during the composition. So it is adaptive when it is elementary and when it is composed. We obtain in the two cases an adaptive web service -elementary or composite-.

The fractal adaptation is to adapt the elementary service in order to obtain an adaptive composite web service. [Laroussi, 2005]

We are interesting in this work to propose a fractal approach of web services adaptation. Our approach is based essentially in two parts: the first one is a proposition of an adaptive description to the web service. The second consist of an adaptive approach to web service composition allowing to obtain an adaptive composite web service. We have base our approach on the context in which we work -a scenario of mobile training- and in the experience of existing adaptation research.

In the following we give an overview of existing adaptation research in order to discuss them. And then we detail the context in which we work, and we conclude with our proposition.

II. ADAPTATION RESEARCH ISSUES
In this section we will outline adaptation -or personalisation- research issues. We are concentrated in researches interesting in the adaptation of Web services primarily considering the composition, the user profiles and the context.

- Adaptation in the web service composition
At present, dynamic Web service discovery and composition is merely driven by technical properties and requirements: activation signatures, interface types, and quality of service characteristics are utilized to gain initial access to service subscriptions and subsequently allow composing and tailoring service offerings.

Research in the area of the Semantic Web seeks a solution to this unsatisfying situation [Berner-Lee, 2001]. Generally, the Semantic Web encompasses efforts to populate the Web with content having formal semantics and rich service descriptions. Recent semantic efforts around UDDI, WSDL and SOAP try to enable automated agents to reason about Web service descriptions and to perform intelligent service discovery, comparison and composition [McIlraith, 2001]. For proactive services behaviour these semantic service descriptions are especially important. During service discovery and selection, decisions that have to be made will not necessarily lead to one definitive outcome. Requirements could be met not by just one service or service component, but by a set of them, or none at all.
At respective decision points during service discovery the discovery process can take the user’s personal preferences provided with his semantically rich personal profile into account. Considering other important context information, an optimal selection and execution of a service is ensured. We present in this paper a composition approach having the goal to obtain an adaptive web service and treating the discovery and the orchestration problem with the same aim (adaptation).

- **Expressiveness of Personal Profiles**

Complex meta data about users, services, components and applications will have to be integrated, modelled and expressed by the means of adequate modelling and profiling languages. For user profiling languages there is a lot to be learned from the database and worlds where the taxonomy or organization of profile elements is often referred to as schema or ontology.

We advocate that the design of future profiling languages for personalization can benefit from the current approaches to the Semantic Web where the layering of content descriptions has a similar quality [Patel, 2002]: On top of XML, RDF [Lassila, 1999] provides a simple yet coherent structure for the expression of basic semantics. The so-called Semantic Web tower then grows more abstract on its next layers towards highly expressive ontology logics, e.g. with the Web ontology language (OWL) [Dean, 2002] which itself builds upon the DARPA Agent Markup Language and the Ontology Inference Layer (DAML+OIL) [Connolly, 2001].

A similar layering concept is also conceivable for the construction of a personal mobile Web and a first step in this direction was already made with the specification which uses RDF to encode schemas and profiles of mobile clients and devices.

While RDF provides a sound basis to state simple user profiles and preferences, higher expressiveness will be fundamental to enhanced user models and truly personal services. What is more, other fundamental knowledge management problems have to be faced, too. For instance the matching of a rich semantic user description against service parameters will lead to a class of problems known as ontology translation and mapping in the area of knowledge management [Gruber, 1993].

- **Usage Pattern and Default Profiles**

Usage pattern and default profiles will play a crucial role in the personalization of mobile services. Efficiency issues have already affected the standardization of a centralized data management and are also important on a semantic level: if a user can be assumed to be of a certain type, i.e. fitting into a certain profile pattern, only a minimal list of attribute overrides have to be transmitted for service execution leading to an efficient use of possibly limited bandwidth.

In addition to efficiency matters the generalization of user profiles into patterns is also crucial for the automatic service discovery and proactive execution. Because even if a common profiling vocabulary is used for user modelling and service specification, there may be missing parameters in the profiles that have to be added by a sensible default pattern. Conversely, if the service request of a user is too specific, i.e. includes too many or unmatchable attributes; his request can be gradually generalized to match a certain service pattern. Besides, service designers typically have specific ideas in which context the service will be applied. These design intentions can best be modelled by adequate patterns related to different user types and contexts. Pattern generalization itself could be achieved by the utilization of meta variables in profiles, the use of generalization policies expressed through additional ontologies, or the definition of generalization rules. Standards methods and commonly used algorithms may be adapted and deployed for service and profile pattern mining. Early work on generalization, e.g. [Dietterich, 1986] [Mitchell, 1982] show the applicability of standard methods that can even be extended to conceptual clustering [Michalski, 1986].

In a further step it might be conceivable that additional patterns exist to describe the user’s intention, situation or context.

**Summary and issue**

In this part we have list a set of research issues in adaptation fields. We have just discuss different problems to resolve in web service adaptation and different solutions proposed by previous research. In our case we try to response to a set of problems essentially adaptation of web services to user model [Laroussi, 2001] and to the context. To do this we will adapt service’s description to these parameters and we will propose a web services’ composition to this purpose. We combine the two approaches, i.e. adaptation of web service’s description and web services’ composition, to obtain an adaptive service.

Our experimental case is a collaborative and mobile scenario of training.

In the following we begin by detailing the context of the current work. We pass after to the adaptation’s approach proposed. We finish by a part of test and validation.

**III. APPLICATION CONTEXT**

Our application context for the adaptation approach proposed is a scenario of collaborative and mobile learning activities.

**A. Mobile Learning**

Mobile learning: It’s learning through mobile computational devices: Palms, Windows CE machines, even digital cell phone.

The vision of mobile computing is that of portable computation with rich interactivity, total connectivity, and powerful processing. This small device is always networked, allowing easy input through pens and/or speech or even a keyboard when necessary (though it may be something completely different like a chord
Fractal adaptive Web service for mobile learning

Keyboard), and the ability to see high resolution images and hear quality sound. It may be that the image is overlaid on the world through glasses that act like a Heads Up Display.

Mobile learning can be considered from two viewpoints. The first one is technical oriented perspective regard traditional behaviouristic educational paradigm as given and tries to represent or to support them with mobile technologies. A main concern from this perspective is how to create, enrich, distribute and display learning material on mobile devices; the main benefits are to personalize the way of learning (where you want, when you want, what you want, as fast as you want, how you want; etc).

The second one is pedagogical socio-cognitive and distributed cognition paradigms. In this viewpoint we face traditional designs of teaching and learning to push community oriented learning like collaborative learning, problem based learning; informal and ad-hoc learning, etc. In this paper, we are interest by both viewpoints. [Laroussi, 2004]

B. The Scenario

In our approach of adaptation we have taken into account a set of general parameters and a set of specific parameters related mainly to the context. The context we choose is a scenario of collaborative and mobile training. We begin with detailing this scenario: it will be useful to clarify how do we proceed in the adaptation approach and how do we test the results.

Various scenarios supporting learning activities are proposed. We chose the Lundin’s [Lundin, 2003] scenario because it treats the collaborative and mobile activities of learning.

The following is a short explanation of this scenario:

The teacher starts by dividing the tasks between students, while explaining the scenario. And then starts a conversation with whether real or virtual learners. Those learners have to use a specific password to be able to participate in the scenario.

The first step mustn’t exceed three hours in which learners finish with more unanswerable questions. In the second step, learners begin to think individually about the solution during no more than a week according to the tasks’ difficulty.

The third step is a seminar which brings together the students and the teacher to discuss the results of the scenario, to connect these results with their daily activities, and finally they become able to identify the problems.

A number of discussions between the participants allows them to find the solution and to release others horizons. The participants divide among them the roles for the future tasks and the reflection is made this time at the base of experiments acquired by the users by the preceding meetings and which make it possible to divide roles them selves, and to continue in the same spirit “to change role successively, to discuss, understand and learn”

Various simulations are generated by the scenario with an aim of ensuring that the participants are in the same scenes of scenario and there aren’t passive participants. The virtual participants (and even others) use mobile devices -PDA, PC, Phone,… to facilitate the discussion with the various participants (student or professor) and so they can easily take part in different phases from the scenario whatever their sites.

The general principle of the scenario is given by the following figure: (numbers indicates the order of the actions in the scenario)

For the beginning we need just some critters of the scenario to release context’s and user’s parameters for the adaptation. But after this scenario will be divided into web services and will be useful for the test.

IV. OUR ADAPTATION’S APPROACH

In our adaptation’s approach we proceed as follow:

In the beginning we adapt the elementary service according to a set of requirements -that we will detail in the paragraphs 4.1 and 4.2-. The adaptation of elementary service concern essentially its description that must take in account adaptation’s parameters and that must allow to the service to be reconfigurable according to these later;

Then we adapt the composition of services, the result is an adaptive composite service. We adapt the composition process and the composite service. The composition process must return to the web service’s description in order to update the web service’s parameters to be adaptive in addition to respond to the composition’s constraints

In the next part we treat each point separately and then we combine them in order to have always an adaptive service.

The following figure explains our approach:
A web service has a syntactic and a semantic description. On the one hand, a syntactic description provides the types of necessary inputs/outputs, and how to write the user request syntactically... On the second hand a semantic description explains semantically the action made by the service.

The web service has also a set of parameters such as the quality of service, the degree of security, the cost, and its availability.

In order to adapt the elementary service we add in the set of inputs others elements concerning the adaptation to the user and to the context such as user’s preferences, user’s level of education, user’s background, user’s cognitive aptitude and user’s learning styles. For the context we add a set of parameters such as mobility, collaboration...etc.

So, to sum up, we can define a fractal adaptive web service as follow:

Fractal Adaptive Web Service = <input, output, task to do, description, parameters, pre-conditions, post-conditions>

Input = set of <events, user’s preference, user’s level of education, background, cognitive aptitudes, learning styles >

Output = set of <events”>

Description = < semantic description, syntactic description >

Parameters = < QOS”, security, availability, cost, value inherited (of the supplier to the receiver), Mobility, adaptability>

Pre-conditions = set of <conditions>

Post-conditions = set of <conditions>

Syntactic Description = < type of the data required...>

Semantic description = < functionality semantically described ...>

Availability = < yes/ no>

Event = < waited result, actions...>

Mobility =< Yes/No, what is mobile>

Adaptation =<change the output according to the user’s preferences and profile and to the context>

The bold text in the specification concern the adaptation’s characteristics added to the classical description of Web service [Tirellil, 2006].

In this level the adaptation relates only to the web service’s description and when we check the service’s adaptation we check the set of the adaptation’s parameters, especially we check that the output change compared to the change of the input and of web service’s parameters.

C. The adaptation in the composition’s service

Composition, Orchestration, choreography, business process management, and workflow are all terms related to connecting web services together in a collaborative fashion. The capabilities offered by web services orchestration will be vital for building dynamic, flexible processes. The goal is to provide a set of open, standards-based protocols for designing and executing these interactions involving multiple web services.

* depend on the set of input, mainly the user’s profile and context’s parameters

** quality of service
Our goal is to obtain a web service which satisfies the user's needs by respecting his preferences, his profiles, his level of academic studies, and the context in which he works...

To do this, we are interested in the adaptive composition -or orchestration- of web services: The result is an adaptive composite service obtained starting from a set of existing services. The choice of these services is made on the base of a set of constraints -which will be detailed after- and the purpose is primarily the adaptation. Therefore, the idea is to use a set of existing web services to form a new composite service adapted to the user’s needs.

We present an adaptive composition as follows:

![Figure 4. The adaptation view in the composition case](image)

A zoom for the surrounded part is:

![Figure 5. The discovery and orchestration with an adaptation's consideration](image)

The web services’ composition is composed of two parts: The first one is the discovery of services: how to find the set of services answering the user’s request in an adequate way. The second part consists of checking the coherence of web services between each other i.e. the problem of orchestration of services chosen -the composition itself-.

The discovery part ends with a checking of the adaptation’s constraints:

- If the selected services are adapted in their descriptions, the orchestration part would be made using the adapted elementary services and the result must be normally an adapted composite service.
- If there is a service, among the selected services, which is not adapted in its description, the orchestration part calls the description’s adaptation of the service concerned.

In all cases the orchestration starts with services adapted in their descriptions and it ends with a checking of the adaptation of the composite service result (it may be that one of the parameters of adaptation changed or that there was an error during the checking of the description’s adaptation of the services selected thus it is the moment to correct it before answering the user’s request).

We propose an algorithm treating the two cases -discovery and orchestration of services-. This algorithm has the purpose to obtain an adaptive composite service.

With the growing number of Web services, importance of composing existing Web services into more complex services in order to achieve new and more useful solutions is increasing. However, composing Web services is not a trivial task especially when talking about dynamically composing Web services at run-time.

Considerable amount of research has been done and currently is underway in this field. A number of frameworks/prototypes using various techniques have been developed by the industry and academia in order to achieve automated composition.

But none of these approaches describes the method adopted for modelling Web services composition and different steps to select and to combine the most relevant Web services. – Those tails to integrate them in the our adaptation approach-

In addition of this we haven’t found an approach having exactly the same purpose we have: composing Web services to obtain an adaptive one in a context of collaborative and mobile scenario of learning. Thus, we propose our own approach of Web service composition in order to respond to our constraints and to have all details we need in the adaptation approach.

The following table summarizes our algorithm in the two cases -discovery and orchestration-:
Fractal adaptive Web service for mobile learning

Table 1*: The Summarized composition’s Algorithm [Tirelli, 2006]

<table>
<thead>
<tr>
<th>1. Discovery</th>
<th>2. Orchestration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1. Determine key words in the request</td>
<td>2.1. Check for conformance for each selected service search which service-in the set of selected services- can be an input</td>
</tr>
<tr>
<td>get request from interface, rewrite it with an abstract method, and retrieve the set of key words</td>
<td>2.2. Define the structure of composition construct the tree structure: each branch is composed of conform services</td>
</tr>
<tr>
<td>1.2. Selecting web services for each stored service if (the description service is all right written)</td>
<td>2.3. Choose the composite services each branch is a service we store it as a composite service for (each composite service) calculate the degree of correspondence keep composite services having a degree of correspondence &gt;=3</td>
</tr>
<tr>
<td>- Apply the context filter: calculate the semantic distances between the keys words</td>
<td>2.4. Check constraints for (each composite service) Check user’s preferences, profile and back ground Check the context’s parameters (mobility and collaboration) Ask user if there is a change in these parameters (user’s and context constraints) if (the service verify the set of constraints) keep it in the selected services else it will be deleted</td>
</tr>
<tr>
<td>Store the selected service in an adequate structure</td>
<td>if (there is no selected service)</td>
</tr>
<tr>
<td>}</td>
<td>{ Write the service description with the abstract method chosen for the request Apply the context filter: calculate the semantic distances between the keys words Store the service selected in an adequate structure for (each selected service in the first filter)</td>
</tr>
<tr>
<td>{ Apply the profile filter: Comparing the words’ occurrences in the service and in the request. Store the selected services }</td>
<td>for (each composite service)</td>
</tr>
<tr>
<td>for (each service selected for the first and the second filter) { Apply the similarity filter Store the selected services }</td>
<td></td>
</tr>
<tr>
<td>2.3. Check constraints Check user’s preferences, profile and back ground Check the context’s parameters (mobility and collaboration) Ask user if there is a change in these parameters (user’s and context constraints) for (each selected service) check the user’s and the context’s constraints if (the service verify the set of constraints) we keep it in the selected services else it will be deleted</td>
<td></td>
</tr>
<tr>
<td>A fractal treatment Check the adaptation for the elementary services Check the adaptation in the composition Finally check the adaptation of composite service 2.5. response to the user request Choice the composite services using a less number of services and having the highest degree of correspondence Store the composite service in the data base of services in order to use it in the future And response to the user’s request</td>
<td></td>
</tr>
</tbody>
</table>

The discovery and the orchestration both form the composition process which is transformed to a web service: the task done is the composition of the services (explained by the algorithm), the input is the user request and a set of existing services and the output is the composite service and the response to the user request.

For the discovery step we choose the more adaptive service to the user’s preferences and profile. In fact, the composition’s service is developed in order to obtain an adaptable response to the user’s request. This service will be used after in the adaptation model that we propose in this work.

For each step in the scenario explained, we use the majority or the totality of the orchestration’s algorithm steps to obtain the more adaptive service. According to the scenario we release adaptation’s constraints especially users’ preferences and profiles.

**Summarized**

We propose in this part a web service adaptation’s approach. This later is made up of two parts: the first one in the elementary service’s adaptation and the second is the composite service’s adaptation and composition’s adaptation. We call this approach a fractal adaptation: adapt the more elementary service in order to obtain an adaptive composite service [Laroussi, 2005].

So the adaptation is treated first for elementary service: we propose a functional description of web service showing what we add to a classical web service adaptation to be adapted—mainly user’s profile and context’s parameters in the service’s input and an output which depend on the input and in the adaptation parameters. Adaptation’s parameters are also added to the web service’s input-. And second in composition process: we take into account for each step in the composition – discovery and orchestration- the adaptation. The result of the composition process is a fractal adaptive web service.

In order to validate our adaptation’s approach, we test it in the following.

**V. TEST AND VALIDATION**

For the test we have implemented the composition’s algorithm and a set of services corresponding to the scenario explained in the beginning of the paper. The validation consists of two parts: the first is the test of services and algorithm separately, and the second concern the integration of services in a platform of mobile learning functioning with respect to the algorithm.

We have implemented the algorithm, a set of web services and the platform.

We consider a population of 15 students. For these later we develop a set of services related to the scenario explained in the beginning, to validate the composition’s algorithm.

**A. Test and validation using a set of services**

A set of web services are defined for the scenario described in the beginning: for the first part of validation, we will test the algorithm with a limited number of these services.

*iJET International Journal of Emerging Technologies in Learning - www.i-jet.org*
We consider for example, the set of web services \{E_1, E_2, E_3, ..., E_n\} recorded in a data base (or in a file) specific to the task of collaborative and mobile.

These services are defined as follows:
E_1, E_{10}, E_{20}, E_{14} are paying (User Profile, portfolio, playing course on PDA, whiteboard).
E_{30} is a reserved service.

\{E_1,...E_4\} is a set of services to manage courses for a weak student.
\{E_5,...E_{25}\} is a set of services to manage courses for an average student.
\{E_{26} and E_{30}\} is a set of services to manage courses for a good student.

\{E_4\} Topic: office, field: Excel.
\{E_5, E_6\} Topic: computer science, field: database.
\{E_5, E_7, E_8, E_9, E_{11}, E_{12}, E_{13}\} Topic: Computer Science, field: Multimedia.
\{E_{15}, E_{16}, E_{17}, E_{18}, E_{19}\} Topic: Computer Science, field: audio-visual.
\{E_{26}, E_{27}, E_{28}, E_{29}\} Topic: Computer Science, field: programming.

For the set of services that we consider, the result of coherence *’s checking is as follows:

![Image](image-url)

Figure 5. The conformance checking (2.1 in Table 1)

In the first column (surrounded in figure 5), for example, the output of E_1 can be an input for E_2. So E_1 is coherent with E_2 but E_2 is not coherent with E_1 because the output of E_2 can not be an input for E_1.

For the example we treat, we consider the follow request -Q-:

Q= Session of mobile collaborative learning for English, computer science and multimedia. This session concerns an average student.

The request and services are written according to the model given in the beginning of the paper. All services are stored in a file or in a data base.

Using the three filters, we defined, the result is:

- All the services answer the constraints of the first filter except the service E_{30} which will be eliminated because it is about a reserved service not an academic service;
- For the second service, and after comparing the occurrence of the words, we assign to each service a degree of correspondence;
- The third filter will compare the similarity of the words. Only the services having a value of Similitude (S1) equal to 1 will be retained with S1 ∈ [0, 1].

The sequences of services selected are in the following table. The first column contains all possible combinations of services that response to the user’s request with respect to his preferences and context in which he works. The second column contains the degree of correspondence of each set of services.

### Table II. Services and degree of correspondence

<table>
<thead>
<tr>
<th>Set of services</th>
<th>( \sum ) of Degree</th>
</tr>
</thead>
<tbody>
<tr>
<td>E_5, E_{10}, E_1</td>
<td>1+1+1=3</td>
</tr>
<tr>
<td>E_5, E_{10}, E_1, E_2</td>
<td>1+1+1+1=5</td>
</tr>
<tr>
<td>E_5, E_{10}, E_1, E_2, E_3</td>
<td>1+1+1+1+1=6</td>
</tr>
<tr>
<td>E_5, E_{10}, E_1, E_2, E_3, E_4</td>
<td>1+1+1+1+1+1=7</td>
</tr>
<tr>
<td>E_5, E_{10}, E_1, E_2, E_3, E_4, E_5</td>
<td>2+2+2+2+2=9</td>
</tr>
<tr>
<td>E_5, E_{10}, E_1, E_2, E_3, E_4, E_5, E_6</td>
<td>2+2+2+2+2+2=10</td>
</tr>
</tbody>
</table>

The best collections of service, or the composite service, that we retain are those having the degree of correspondence \( \geq 3 \). If there are two collections having the same degree we retain the collection having the less number of services and which is more adaptive.

The algorithm provides the composite service not the set of separate service which composes it.

The first part of the test is made theoretically by using a questionnaire, and taking in account students’ answers we made turn the algorithm.

In the next part an implementation of a platform of mobile training is made, the algorithm is integrated in the platform and this latter functions by respecting the already detailed scenario.

### B. Test and validation using a platform for mobile learning based on web services

Once the algorithm is tested with a set of Web services, it would be used in a collaborative platform for mobile learning based on fractal adaptive e-services [Tirellili, 2005].

For this step of test a set of web services are implemented -according to the scenario proposed- given in the following figure:

^ two web services are coherent if the output of one can be the out put of the other

* The platform is already developed -Mai 2005-
In this step we only implement the bold services, explained as follow:

- **Security and Identification services**: It’s a category of access rights management, security and identification. This category contains essentially: Logging, Digital Rights Management, Authentication, Authorisation…

- **Help services**: This category is dedicated to the various documents research and assistance. It’s containing essentially: Search, Federated Search, Research, Resolver, Terminology…

- **Services for Learning system**: In this category, we find services dedicated to activities' learning (Activity Author, Activity Management, Course Management, Grading, …)

- **Personalization / Adaptation services**: This category contains all necessary services for having a current learning session. It concerns the adaptability of content. (Learner profiles management, User Preferences, Context, Format Conversion, Metadata Management, Metadata Schema Registry, Tracking, Recording and extraction of learners' performance information…)

- **Evaluation services**: These groups of services allow the management of contents and the teaching sequencing of training activities such as evaluations.

- **Collaboration and Mobility services**: This category includes services allowing the creation and the management of sessions of collaboration both for synchronous and asynchronous communications (Alert, Chat, Forum, Whiteboard, Messaging…).

The selection of services used in this step is referenced in [Wilson, 2004]. These services are integrated in the platform SOLEIL (Service Oriented Layered Environment for Interactive Learning), an interface adapted to each class of users –Administrator, Student and Tutor- is also developed.

An example of using the platform and the result of adaptation approach is explaining in the following:

- The first step for all users is the authentication step. According to the result of this step an appropriate window is open for each user:

- The administrator can consult his notes, delete some of them or add others notes;

- The administrator have the right to manage users: add a student to an appropriate group according to his education level, his profile, his preferences and the context in which he works:

- The administrator can add a tutor for an appropriate group according to the adaptations parameters:
VI. CONCLUSION

This paper describes our approach for realizing personalized web services. Our approach is driven by the goal of realizing an approach which allows a user to select, customize and combine personalization functionality. To achieve this goal, we have developed an approach for creating and maintaining personalization services, the fractal adaptation approach. This approach provides an environment for accessing, invoking and combining personalization services, and contains a flexible, service-based infrastructure for visualizing adaptation outcomes, and for creating the user interface.

Up to now, we have realized a platform for mobile e-learning based on adaptable web services in order to check the approach proposed. Currently, we are implementing further personalization services (listed in figure 6), extending the user model and transforming it on web services.

REFERENCES

Fractal adaptive Web service for mobile learning

AUTHORS

Ichraf Tirellil is with Laboratoire Trigone - CUEEP - Bât B6 Université des Sciences et Technologies de Lille 59655 Villeneuve d'Ascq cedex – France and RIADI ENSI campus de la Manouba 2010 Tunisia (e-mail: Ichraf.tirellil@riadi.rnu.tn)

Mona Laroussi is with Laboratoire Trigone - CUEEP - Bât B6 Université des Sciences et Technologies de Lille 59655 Villeneuve d'Ascq cedex – France and INSAT Centre Urbain Tunis Nord BP 676 CEDEX 1080 Tunis, Tunisia (e-mail : Mona.laroussi@univ-lille1.fr).

Alain Derycke is with Laboratoire Trigone - CUEEP - Bât B6 Université des Sciences et Technologies de Lille 59655 Villeneuve d'Ascq cedex – France and INSAT Centre Urbain Tunis Nord BP 676 CEDEX 1080 Tunis, Tunisia (e-mail : Alain.derycke@univ-lille1.fr).

Henda BenGHezala is with RIADI ENSI campus de la Manouba 2010 Tunisia.

Manuscript received 12 May 2006.


