ISSN 2047-3338

Detection of Preference and Selection of Cloud Services for Dynamic Adaptation Based on the User Profile

KANGA Koffi[1], GOORE Bi Tra[2], BABRI Michel[3] and OUMTANAGA Souleymane[4]

[1,2,3]Ecole Doctorale Polytechnique de l'Institut Nationale Polytechnique Félix Houphouët Boigny (EDP/INPHB), Côte D'ivoire / Laboratoire de Mathématiques et des Nouvelles Technologies de l´Information (LMNTI)

[4]Laboratoire de Recherche en Informatique et Télécommunication (LARIT)

[1]kangamiage@hotmail.com, [2]goore@inphb.edu.ci, [3]michel.babri@inphb.edu.ci, [4]oumtana@gmail.com

***Abstract*– In this article, we propose a user's preferences detection tools in cloud computing and also the selection of services available on request by the cloud providers, so as to make a dynamic adaptation of these services to the user's preferences according to their profile. This contribution focuses first of all on the implementation of data models and then on the different algorithms allowing the detection based on the proposed models. Therefore, the different algorithms were developed according to the different formalizations set up, namely: Matrix formalization in detecting user's preferences based on their profiles, and Matrix formalization for selecting cloud services based on user's profile.**

***Index Terms*– Cloud Computing, Dynamic Adaptation of Services, User Profile, Preference Detection and Services Selection**

I. INTRODUCTION

TALKING about user's preference and cloud services detection is something that matters a lot. It therefore needs a little explanation. In fact , cloud computing is a service or a concept that consist to deploy on remote servers data processing traditionally located on local servers or on the user's desktop .

It is an IT services in the form of on request service, accessible from anywhere, anytime and by anyone with an authentication system. Thus, it involves two (2) major actors:

- Service provider offering on-request services;
- End user who consumes a service

All the services offered can be divided into three categories:

- IaaS (Infrastructure as a Service)
- PaaS (Platform as a Service)
- SaaS (Software as a Service)

As for the adaptation, it is an adjustment of services offered by the providers of Cloud in respect of the preferences and tastes of the user. It allows him to enjoy and see how useful the Cloud services are for him.

In such a context, adaptation becomes important for the client, because it will allows him to find satisfaction during and after consumption also it enables the service provider to keep in touch with its customers.

Faced with this dilemma, several research studies have tried to find solutions regarding expressiveness profiles and adaptation process of multimedia content [1], supported platforms and adaptation [2], [3], adapting the service of Graphic User Interface (GUI) [4], adapting applications and to the context using web services.

In order to make our contribution to the adaptation work to satisfy both parties (client and service provider), we suggest two(2) data models, their formalization and algorithms to identify user's preferences but also the services offered in order to make an adaptation of these two elements in the consumption of services by customers.

The rest of the article is organized as follows: Section 2 develops a state of the art in this field. Section 3 state the problem. Section4 makes room for our contribution. Section 5 discusses the limitations of the work contribution, by presenting the benefits of cloud service to service providers and users. Section 6 rounds up with a conclusion and some perspectives.

II. STATE OF THE ART

A) Dynamic adaptation of service [3]

The objective of this study is to develop an architecture for the adaptation of cloud computing services. It includes three parts:

- A Modifiable part: including the components and interconnections between them
- A Monitoring part: including elements observing resources and user profiles.

They all provide useful elements to the description of the service called meta-description

- A Control part: represented by the adapter and an assembler which respectively allows to decide on the changes to be made to adapt the service and who should performs the action requested by that the adapter.

The experimental setting chosen by the authors of this architecture is that of a forum service originally created for English users. This architecture helps adapt the translation service to non-English Speakers or Users as well.

Limits

The limitations of this study lies in its complexity.

The architecture ignores semantic service composition whose realization requires the comprehension of functioning of the set of “service-context” by the machine as well as human.

It has not been possible to realize it in a cloud environment.

B) Adaptation to the web using context of applications [2]

This work is the result of a project called SECAS (Single Environment for Context Aware System) interested in adapting the applications to the context of use. The objective is to establish a platform making applications adaptable to different contexts of use which are data, services and presentation.

To do this an architecture was proposed as part of this project implying the definition of the context proposed by Dey [2] and includes:

- A context manager (context sensor, context interpreter and storage of the historical context)
- An adaptation system (consisting of an adapter composed of a user interface adaptation service, content adaptation service and adaptive behavior services)
- Basic applications composed of basic services, data and user interface.

Also in order to make the applications adaptable, the authors set up in their work some operators to adapt behavior, content adaptation tools and a process of adaptation of the presentation through web services.

Limits

This work has numerous limits. Among which are:

- Lack of detection tools for profiles and user’s preferences
- Non consideration of environments and cloud computing platforms where three forms of services (IaaS, PaaS and SaaS) are offered to users.
- The independency of services.

C) IDM approach to ergonomic adaptation to the mobile HMI context [4]

This study seeks to overcome the inconsistencies between the formats of presentations (User Interface). To do this, a research based on MDE approaches (Model Driven Engineering) [5] (for specifying a system regardless how of the platform that hosts) it was made.

So in this paper, the authors proposed a MDA (Model Driven Architecture) for the automatic generation of context-adaptable UI .This approach has enabled the implementation of user’s context models, environmental context models and platform context models. These different models have generated an HMI adaptation process at three levels starting with an independent context model.

Limits

Although, the GUI adjustment is part of the solution, this work does not take into account the instantaneous needs of the user and also was done in a ubiquitous computing environment where a user engages in his activities several computing devices and systems. The achievement was not made in a Cloud.

D) A Framework for dynamic resource provisioning an adaptation in IaaS clouds [6]

IaaS provides the ability to acquire and free up IT resources on request to dynamically adapt the workload.

The aim of this work is the simultaneous minimization of the costs of resources and runtime services within a Cloud infrastructure to achieve adaptation to the growing of user’s needs of resources.

So, in this article, the authors propose a framework for resources supply and adaptation. The center of this framework is a set of adaptive algorithms able to make clear decisions to adjust resources according to workload.

This algorithm is designed to manage all acquired resources among multiple cloud providers geographically distributed so that they interact with local resources.

Limits

The limits that presents this work to our knowledge are followings:

- The lack of consideration of user context to identify the profile and services, which are consumers of resources.
- Lack of identification of cloud services that are the basis of the increase in the workload of IaaS resources
- As this work does not specify what type of resource it is talking about (CPU, RAM, etc. ...)

E) Towards an automatic adaptation of heterogeneous multimedia mobile application [7]

This work was conducted as part of ubiquitous computing where application execution environment is consisted of heterogeneous elements such as PDAs, laptops, smartphones etc. composed with heterogeneous material resources. These objects belong to users having varied needs. Given these parameters, it is appropriate to structure application softwares entities called independent components so as to facilitate their adaptation.

The proposal in this work consists in the automatic semantic adaptation based on the use of ontologies. It then aims to highlight the adaptation needs in one hand, and to facilitate the search and identification of the other services in order to implement the necessary changes.

For the realization of this proposal the following objectives were set:

- the definition of an ontology describing the semantic information of multimedia adaptation services (name, location, category, role ...) and the parameters for QoS adaptation services. (Average of execution time, transfer time and the quality percentage)
- the definition of an inference engine responsible for optimizing and adaptation planning to implement on a media server
- The use of web services to perform the adaptation.

For construction of adaptation services, an algorithm was proposed. It takes as input the profile description and adaptation resources to output appropriate resources. The steps of this algorithm are the following:

- The initialization of the automatic adjustment by the inference engine
- Seeking ways of adaptation
- Selecting the best paths
- Extraction adaptation services
- The implementation of the adaptation

Limits

The limitations noted in this work, include:

- The improvement of the ontology which is used for the representation of the quality of sensitive services to the context and complete the proposed approach based on SOA services which are suitable for cloud implementation.
- The consideration of cloud characteristics in the semantic adaptation process.

III. PROBLEM

The above review of existing work reveals that most authors performed excellent work in the framework of the adaptation of service. However, very few of them have oriented their work in the field of dynamic adaptation service to the profile and cloud user preference apart Nguyen Binh Ta and Al. in [6]. In fact, some author have based their work on the implementation of adaptation services architecture [1] in the traditional web environments.

Others however just use the web services for the implementation of software adaptation platform regardless of the dynamicity need and preferences of users [2], much less cloud computing services since this work is not known to cloud-based experimental setting.

As for Dorra ZAIBI & Al. they are oriented in the search of solutions for the adaptation of ergonomics in a GUI environment for mobile [4].

BACHA [8], [9] meanwhile, offers adaptive solutions based on content personalization.

That said, none of the research has raised the issue of profiles detection and cloud user preferences detection because there were few who were oriented to Cloud. Also, the detection of cloud services based on user profiles has not emerged.

So, we will try to propose an approach to highlight cloud users, their profiles and preferences and the various cloud services to initiate appropriate adaptation based on algorithms manipulating data models we have developed.

IV. CONTRIBUTION

A) Preferences detection according to the user profile

Our contribution is based on three (3) points:

- Setting up of a data model for detection (Fig. 1)
- Formalization of the detection
- Implementation of a detection algorithm

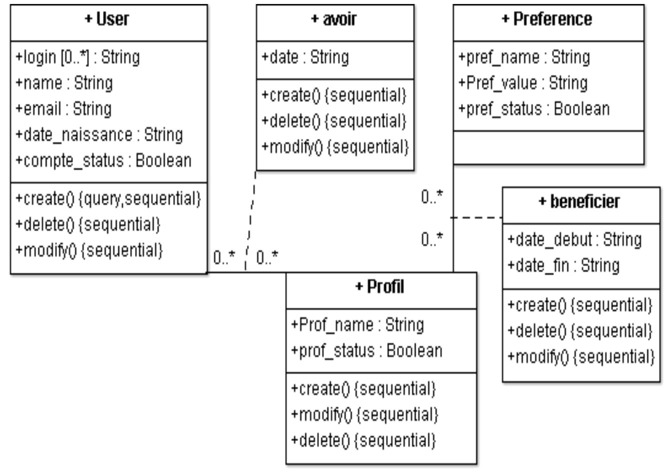

Fig. 1 : class diagram for the detection of preference according to the profile

1) Model of detection

The proposed model includes the following classes:

- *User: The* use of cloud service requires user authentification. So, a number of information is needed. We materialize those - one through the class« User ». This class includes the following attributes: login, name, email, and date of birth and the status of the account «compte_status».
- *Profil:* in the cloud, users are classified by profile (eg the guest profile, director, etc. ...).Each profile is characterized by a name (prof_name) and status (prof_status) which is boolean type.
- Avoir: this class is an association class. It enables to materialize the date on which the user has selected a profile through the attribute« date ».
- *Préférence:* preference in the Cloud refers to a set of elements related to the taste of the user, in this regard, if the user prefers a given web service to another, then the materialization of this preference is made through the attributes "pref_name" to the name of the preference, "pref_value" for the value of this preference and "pref_status "to the status (on / off) of it.
- *beneficier: in* this class, we materialize all links between profile and preferences. These links allow any user to know the times (start date and end date) in which the user show some interest in preferences through the profiles that may vary over time.

2) Formalizing preferences detection

Formally speaking, the detection of user preference is made up of three sets. These are:

- $U = (U_1, U_2, \ldots U_n)$ all users or customers of the Cloud
- $P = (P_1, P_2, \ldots P_k)$: all profiles defined in the Cloud
- $Q = (Q_1, Q_2, \ldots Q_m)$: a set of all defined preferences

We define the matrices as follows:

UP (U_i, P_j) with $(1 \le i \le n,\ 1 \le j \le k)$. (n is a maximum of user and k is a maximum of profile)

PQ (P_j, Q_x) with $(1 \le j \le k,\ 1 \le x \le m)$. (m is a maximum of preference)

UQ (U_i, Q_j) with $(1 \le i \le n,\ 1 \le x \le m)$.

$$UP(Ui, Pj) = \begin{cases} 1 \ if\ Ui\ has\ the\ profile\ Pj \\ 0\ otherwise \end{cases}$$

$$PQ(Pj,Qx)=\begin{cases}1\ if\ Pj\ contains\ preference\ Qx\\ 0\ otherwise\end{cases}$$

$$UQ(Ui,Qx)=\begin{cases}1\ if\ \big((UP(Ui,Pj)=1)et(PQ(Pj,Qx)=1)\big)\\ 0\ otherwise\end{cases}$$

Example:

Let U, P and Q be a set of items with

U = {U_1, U_2, U_3, U_4, U_5} a set of five (5) cloud users

P = {P_1, P_2, P_3, P_4} a set of four (4) profiles and

Q = {Q_1, Q_2, Q_3, Q_4, Q_5, Q_6} a set of six (6) preferences

$$UP\ (Ui,\ Pj)=\begin{pmatrix}1&0&0&0\\0&1&1&0\\1&0&0&1\\0&1&0&1\\1&0&1&0\end{pmatrix}$$

The rows of the matrix denote (U) a set of five (5) users and the columns denote (P) a set of four (4) profiles.

With

$1\leq i\leq 5\ and\ 1\ \leq j\leq 4$ Where (i, j) are integers

And

$$PQ\ (Pj,\ Qx)=\begin{pmatrix}1&0&0&0&0&0\\1&0&0&1&0&1\\1&0&0&1&0&1\\0&1&0&1&0&0\end{pmatrix}$$

The rows of the matrix denote (P) a set of four (4) profiles and the columns denote (Q) a set of six (6) preferences.

With

$$1\leq j\leq 4\ and$$
$$1\ \leq x\leq 6\ \ where(j,x)are\ integers$$

So by inference we have:

$$UQ\ (Ui,\ Qx)=\begin{pmatrix}1&0&0&0&0&0\\1&0&0&1&0&1\\1&1&0&1&0&0\\1&1&0&1&0&1\\1&0&0&1&0&1\end{pmatrix}$$

The rows of the matrix denote (U) a set of five (5) users and columns denote (Q) a set of six (6) preferences.

With

$$1\leq i\leq 5\ and$$
$$1\ \leq x\leq 6\ \ where\ i,x\ are\ integers$$

In our example the application of our model provides the following preferences:

- Q_1 is preferred by user U_1 ,
- Q_1, Q_4 and Q_6 are preferred by user U_2
- Q_1, Q_2 and Q_4 are preferred by user U_3
- Q_1, Q_2, Q_4 and Q_6 are preferred by user U_4
- Q_1, Q_4 and Q_6 are preferred by user U_5

3) Proposed detection algorithm (Fig. 2)

Our proposed algorithm allows:

- To detect the profiles to which a user belongs on the basis of knowledge of the information supplied during access authentication to the Cloud. The input is the set U of users.

Also to detect a set of preferences (Q) associated with the set of profile (P). For each profile, the algorithm detects the various elements (Qx) associated with a profile (Pj).

```
Algorithm1 : detection of preference
Input : U = set of user
Output : Q'= set of detected preferences =∅
Begin
// route the set of all users
For each Ui∈ U do
    // route the set of all profile
  For each Pj∈ P do
// test if Ui have the profil Pj
     If (Ui, Pj) = 1 then
         // routing of the set of all  preferences
                For each Qx ∈ Q do
                      If (Pj, Qx) = 1 Then
                                  Q' = Q'+Qx
                      end If
                      Write ("User " Ui " have the
                      preference " Qx)
                end for
      end if
   end for
         write ("User" Ui " have a set of all preferences "
         Q')
end for
End.
```

Fig. 2: Algorithm for preferences detection

B) Model for services selection according to the user profile

In addition to the basic services offered by the cloud provider for the default profiles, selection allows the user to make its adaptation needs to adjust the functions of a service A to those of a service B. This form of adaptation is indeed under several constraints which include the maximization services while keeping constant IaaS storage space offered by the supplier; all this based on the user profile.

For the management of this selection, we set up the data model in Fig. 3. The different classes in this model are:

- *profile:* identical to that of Fig. 1
- *service:* it represents the SaaS or PaaS services installed on a cloud infrastructure.
- *Infrastructure:* represents the physical characteristics (CPU, RAM, hard drive size and cost per hour of use) of an infrastructure.
- *Consommation:* materializes track of selected services consumption. It is characterized by the consumption time (time consumption), the cost of consumption (cost) and the functioning status (on or off) of the service for the profile associated with it.

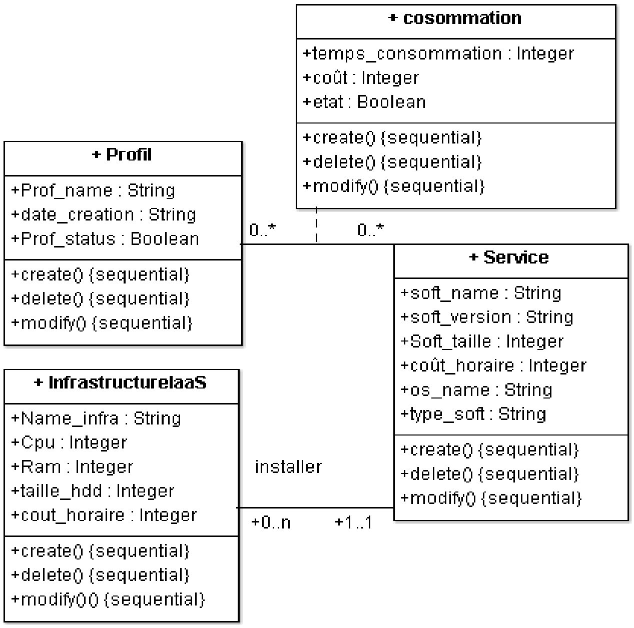

Fig. 3 : Model for services selection according to the user profile

C) Formalizing of services selection according to the user profile

Our formalization of the selection of services is in its form identical to that of the profiles detection. Namely: the use of matrices whose values belong to the set {0; 1}.

In fact, we use the set of profile P = (P_1, P_2,…P_k). As for the other elements involved in the formalization, they are as follows:

- Let S be a set composed as , S= {S_1, S2, S3,…,S_d) of d services proposed by a supplier
- Let I be = {I_1, I_2, I_3…Im) a set of m infrastructures hosting various cloud services.

1) Matrix of SaaS or PaaS service selection:

Our matrix of services selection is formalized as follows:

$$PS\ (Pi, Sk) = \begin{cases} 1\ if\ the\ profile\ Pi\ use\ the\ service\ Sk \\ 0\ otherwise \end{cases}$$

Example:

Let P be = (P_1, P_2, P_3, P_4) a set of four (4) users of cloud and S = (S_1, S_2, S_3) a set of three (3) SaaS or PaaS Services .In these conditions the matrix

$$PS\ (Pi, Sk) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

On this matrix, the rows denote a set of four (4) profiles and columns denote a set of three (3) cloud services.
With

$$1 \le i \le 4\ and$$
$$1\ \le k \le 3\ \ with\ i, k\ are\ integers$$

2) Hosting matrix of SaaS or PaaS services:

Here, the hosting service allows to know the services managed on IaaS platform, in matrix form, it is formalized as follows:

$$SI(Sk, Ip) = \begin{cases} 1\ if\ \ Sk\ is\ hosting\ on\ Ip \\ 0\ \ otherwise \\ with\ \ k, p\ being\ integers \end{cases}$$

Example:

$$SI\ (S_k, I_p) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

Here, the lines denote the set of the three (3) above-mentioned services and the columns denote a set of three (3) infrastructures.
With

$$1 \le k \le 3\ and$$
$$1\ \le p \le 3\ where\ k, p\ are\ integers$$

In this example, we have a matrix which included the following facts:

- the infrastructure I_1 host the service S_1
- the infrastructure I_2 host also the services S_2 and S_3
- as for the infrastructure I_3 it does not host any services

3) Matrix of Infrastructure selection

This matrix is obtained by the association of SI and PI matrix. It is formalized as follows:

$$PI\ (Pi, Ip) = \begin{cases} 1\ if\ (PS(Pi, Sk) = 1)\ and\ (SI(Sk, Ip) = 1) \\ 0\ otherwise \\ and\ where\ i, k\ , p\ are\ integers \end{cases}$$

Example:

$$PI = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

This matrix is obtained by deducting the PS and SI matrix. The rows denote P a set of four (4) profiles and columns denote I a set of three cloud infrastructures.

$$with$$
$$1 \le i \le 3\ et\ 1\ \le p \le 3\ \ where\ k, p\ are\ integers$$

On this matrix we have a set of four (4) profiles that use IaaS services of three (3) infrastructures.

They are as follows:

- The profile P_1uses the infrastructures I_1andI_2
- The profile P_2uses the infrastructure I_2
- The profile P_3uses no infrastructure
- The profile P_4 uses only the infrastructure I_1

4) Service selection Algorithm

Our service selection algorithm takes as input the set S of all services (SaaS or PaaS), the set P of all profiles and comes in four (4) steps. These steps are:

Step1:
Select a profile either SaaS or PaaS service

Step2:
Check if the profile uses the service in question
If yes then add the service to all sets of S services this profile uses.

Step3:
For all services belonging to the set S ', look up for supported the infrastructure

Step4:
After step 3, if we find an infrastructure supported service then it is added to the set I of infrastructure supporting a workload.

These steps are summarized in Algorithm 2 (Figure 4) as shown below:

Algorithm2 : selection of SaaS or PaaS service
Input : P = a set of user profile
S = a set of SaaS or PaaS services
I = a set of the cloud provider's infrastructure
Output : S'= a set of services for each user profile
I' = a set of infrastructures hosting the SaaS or PaaS services

---

Begin
// selection of SaaS or PaaS services
for each Pi $\in Pdo$
for each Sk $\in Sdo$
if (Pi,Sk) = 1 *then*
S' $\leftarrow S' + Sk$
end if
end for
write (" the profile " Pi " have the set of services " S')
end for
// selection of IaaS infrastructures
for each Sk $\in S'do$
for each Ip $\in Ido$
if (Sk,Ip) = 1 *then*
I' $\leftarrow I' + Ip$
end if
end for
write (" the service " Sk " is hosted by" I')
end for
write (" the profile " Pi " uses the resources of infrastructure " I' "through its services" S')
End.

Fig. 4 : Algorithm of services selection

V. DISCUSSION

Dynamic adaptation of the service to user context in a Cloud computing requires the identification of contexts and services.

In our case, the concerned contexts are profile and preferences of the user.

As for the cloud service, we have firstly focused our work on SaaS or PaaS service and in a second place have deduced those of IaaS.

The analysis of the proposed models and their formalization show that they help to detect or to select profiles or services to implement an adaptation strategy for user's satisfaction.

Moreover, these selection operations may allow an assessment of the volume of occupation of these services on an IaaS infrastructure. This assessment is a size indicator services .The user can then adjust or adapt these services. In fact, in an overrunning context, the Cloud may generate a financial extra cost for the user.

VI. CONCLUSION

The aim of our work is to create a framework for the identification and selection of users, their profiles and preferences in terms of cloud services for an adjustment of its services based on user profiles. To do this, we have set up data models for managing profile detections and user preference but also the selection of cloud services to be adapted. We also proposed a formalization of these detection and selection models.

Our work is of course opened. In the future we plan to analyze the management of overrun and space occupation at the IaaS.

REFERENCES

[1] Alti Adel, Laborie Sébastien, Roose Philippe, 'Enrichir l'expressivité des profils et des processus d'adaptation de contenus multimédias', Document numérique 2/2014 (Vol.17), p. 101-124 URL: www.cairn.info/revue-document-numerique-2014-2-page-101.htm. DOI: 10.3166/DN.17.2.101-124.

[2] Chaari, T., Laforest, F., & Flory, A. (2005, May). Adaptation des applications au contexte en utilisant les services web. In Proceedings of the 2nd French-speaking conference on Mobility and ubiquity computing (pp. 111-118). ACM.

[3] Cremene, M., Riveill, M., Martel, C., Loghin, C., & Miron, C. (2004). Adaptation dynamique de services. arXivpreprintcs/0411056.

[4] Dorra ZAIBI et al. Une approche IDM pour l'adaptation ergonomique au contexte d'IHM mobile. (2012) Ecole Supérieure des Sciences et des Techniques de Tunis 5, av. Taha Hussein, Montfleury, 1008 Tunis-Tunisie zaibi.dorra@gmail.com

[5] MDA, Model Driven Architecture. http://www.omg.org/mda (accessed in June 2015)

[6] Duong, T. N. B., Li, X., &Goh, R. S. M. (2011, November). A framework for dynamic resource provisioning and adaptation in IaaS clouds. In Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on (pp. 312-319). IEEE.

[7] Alti, A., Roose, P., Saffidine, R., &Laborie, S. (2012, October). Towards an Automatic Adaptation of Heterogeneous Multimedia Mobile Applications. In NOTERE/CFIP 2012 (pp. 79-87). Cépadues Editions.

[8] Bacha, F., Oliveira, K. and Abed, M. A Model Driven Architecture Approach for User Interface Generation Focused on Content Personalization. In Proc.IEEE RCIS, (2011).

[9] Bacha, F. (2013). Une approche MDA pour l'intégration de la personnalisation du contenu dans la conception et la génération des applications Interactives (Doctoral dissertation, Université de Valenciennes et du Hainaut-Cambresis).

[10] Soukkarih, B., &Sedes, F. (2009) Adaptation dynamique des services au contexte de l'utilisateur.

[11] Ferry, N., Lavirotte, S., Rey, G., &Tigli, J. Y. (2008). Adaptation Dynamique d'Applications au Contexte en Informatique Ambiante. Rapport Technique, Laboratoire I3S (Université de Nice Sophia Antipolis/CNRS), numéro I3S/RR-2008-20-FR, Sophia Antipolis, France.