

SUPPORT DE L'ACTIVITE COOPERATIVE DANS UN CADRE MOBILE

Yvan Peter, Thomas Vantroys, José Rouillard

Laboratoire TRIGONE – Equipe Noce

Université des Sciences et Technologies de Lille

Bât. B6 – Cité Scientifique – 59655 Villeneuve d'Ascq

{Yvan.Peter, Thomas.Vantroys, Jose.Rouillard}@univ-lille1.fr

Résumé — Le projet PLACE (PLAteforme à Composant Evolutive) vise à fournir un cadre pour le support à l'activité coopérative. La plate-forme est bâtie autour d'un moteur de Workflow flexible nommé COW (Cooperative Open Workflow) qui permet d'animer des processus dans des champs d'applications de type enseignement à distance et commerce électronique. Le moteur de Workflow permet aux utilisateurs de connaître les tâches qui leurs sont allouées et d'accéder aux outils et documents nécessaires. Les champs applicatifs visés nous ont amenés à travailler sur des solutions offrant aux utilisateurs un accès via différents types de matériels afin de supporter la mobilité. Nous décrivons dans cet article ce travail basé sur les technologies XML /XSL.

1. Introduction

Les environnements de travail coopératif permettent à un groupe d'utilisateurs d'effectuer un travail en commun. Pour cela, ils doivent offrir des moyens de partage des artefacts manipulés, de coordination des actions des utilisateurs et de communication. La tendance pour ce type d'environnements est de fournir un accès web et l'évolution des technologies de la mobilité incite à les étendre vers d'autres types de supports (PDA, téléphone portable, ...). Le projet PLACE (PLAteforme à Composants Evolutive) vise à fournir un environnement de travail coopératif basé sur des technologies à composants. Le cœur de la plate-forme est un moteur de Workflow nommé COW (Cooperative Open Workflow) qui permet aux utilisateurs de connaître les tâches qui leurs sont allouées et d'accéder aux documents et outils nécessaires. Les champs d'application visés sont l'enseignement collaboratif ainsi que le commerce électronique. La plate-forme utilise un cadre de publication XML permettant de fournir une interface client « universelle » et personnalisée permettant aux utilisateurs d'accéder à leur environnement à tout moment et avec des matériels multiples. Le plan de l'article est le suivant : la première partie décrit brièvement la plate-forme PLACE dans son ensemble ; la partie deux porte sur l'utilisation du langage XML pour fournir un accès « universel » ; la conclusion nous permettra de faire le bilan de notre approche et de tracer les perspectives.

2. La plate-forme PLACE

Le projet PLACE vise à réaliser une plate-forme flexible pour le travail coopératif et l'enseignement collaboratif. Nous avons choisi d'utiliser le modèle de composants Enterprise JavaBeans (EJB) [4] et l'environnement Java 2 Enterprise Edition (J2EE) comme base de nos développements en suivant une architecture multi-niveaux. Sur cette base, nous avons développé un ensemble de services comme la gestion des utilisateurs et des groupes, la communication de groupe, le Workflow ainsi que des outils de travail coopératif (cf. Figure 1).

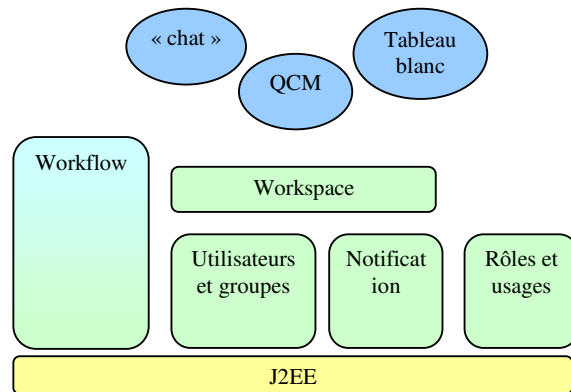


Figure 1 – La plate-forme PLACE

Les services de la plate-forme ont été développés sous forme de composants EJB. L'accès aux services se fait exclusivement à travers des sessions EJB. Ces sessions renvoient des données au format XML à la couche présentation afin de permettre des transformations multiples (par ex. en fonction du type de matériel et du rôle de l'utilisateur). La couche présentation est basée sur le projet Apache Cocoon [3] qui fournit un cadre de publication XML. Cette approche client léger nous permet de prendre en compte les périphériques suivants :

- Ordinateurs personnels (PC/MAC) qui possèdent de nombreux moyens d'interaction (clavier, souris, ...) et de présentation (texte, images, sons, ...)
- Téléphones portables qui peuvent permettre une interaction de type « web » à travers le protocole WAP (Wireless Application Protocol) et le langage de présentation WML (Wireless Markup Language) ou une interaction vocale. L'interaction vocale peut être décrite avec un langage de type VoiceXML [8]
- Téléphones fixes pour lesquels les interactions sont limitées à la voix et au clavier.

La gestion de la présentation est répartie sur un serveur de type web d'une part et sur un serveur vocal d'autre part. Les serveurs de présentation utilisent les données fournies par le Workflow au format XML.

3. Utilisation d'XML pour gérer les périphériques multiples

Notre motivation pour utiliser XML est de pouvoir effectuer de multiples transformations pour adapter les données et la présentation à différents facteurs tels que le rôle de l'utilisateur ou le type de périphérique utilisé. Il est nécessaire de pouvoir exprimer le comportement de la couche présentation en terme de transformation et d'agrèger des données issues de la couche traitement avec des informations de présentation. Pour cela, nous avons utilisé le cadre de publication XML Cocoon fourni par Apache. Dans ce cadre, nous avons développé les éléments permettant d'accéder à la couche traitement (i.e., les EJB session) pour récupérer des données (par exemple les activités à réaliser). Nous décrivons dans la suite le travail portant sur la description de la présentation qui permet de formater les données obtenues.

3.1. Les langages de représentation abstraite

Dans le domaine des Interfaces Homme-Machine, l'apparition d'une multitude de matériels (téléphones portables, PDA, WebTV, ...) ayant des capacités variables appelle à une réflexion sur des techniques de spécification d'interface indépendante de ces matériels afin d'éviter des développements spécifiques [5][6][2]. Cette tendance s'illustre aussi par l'apparition de langages tels que UIML (User Interface Markup Language) [7] qui permettent de décrire une interface. Cette description permet dans un deuxième temps de générer des interfaces adaptées aux matériels. Toutefois ces langages restent proches des modes d'interaction graphique ce qui limite leur application vers de nouveaux modes d'interaction tels que la voix. La suite de l'article illustre le travail réalisé à travers un exemple.

4.1 Application à différents modes d'interaction

L'exemple suivant illustre une description abstraite de page de login (figure 2). Celle-ci peut être utilisée pour générer les interfaces représentées figure 3 ou une interaction vocale. Cette génération est réalisée lors de l'accès à la ressource au moyen d'une feuille de style adaptée au type de matériel.

```
<form Action="http://localhost/verif"
Method="POST" Name="welcome">
<field Type="text" Description="login"
Length="8" Max="10" value="emma"
Name="ask_login"
Voice_Grammar="login_user.gram"/>
<field Type="secret" Description="votre mot de
passe" Length="6" Max="6" Name="ask_pw"/>
<field Type="reset" Name="To_Cancel"
Value="Annuler"/>
<field Type="submit" Name="To_Submit"
Value="Valider"/>
</form>
```

FIGURE 2 – Description abstraite d'IHM



FIGURE 3 – IHM générées

Dans le document XML décrivant la présentation, certaines balises vont être transformées dans une représentation adaptée au langage cible (HTML, WML ou VoiceXML).

Par exemple, le champ mot de passe étant de type « secret », il sera transformé en un champ masqué, lors d'une interaction graphique, alors que lors d'une utilisation avec un téléphone, la saisie se fera au clavier (DTMF) plutôt que vocalement.

4. Conclusion

L'apparition de matériels multiples permet à l'utilisateur d'accéder à des services dans les situations les plus variées et en tout lieux. Dans le cadre du développement de la plate-forme PLACE, nous avons été amenés à travailler sur les techniques permettant d'adapter une présentation abstraite de l'interface utilisateur pour ces différents matériels. Afin de pouvoir gérer des modes d'interactions divers, nous nous sommes détachés des représentations trop proches de l'interaction graphique afin de gagner en abstraction et en réutilisation. On peut distinguer les éléments liés à l'interaction avec les utilisateurs (saisie d'information) et les éléments liés à la présentation des données issues de la couche traitement (tableau, graphique, ...). Il reste encore à généraliser et systématiser les règles d'adaptation de ces éléments de présentation aux capacités des matériels.

Bibliographie

- [1] Anderson, E. A., Breitenbach, S., Burd, T., Chidambaram, N., Houle, P., D. Newsome, D, Tang, X., Zhu, X., Early Adopter VoiceXML, Wrox, 2001.
- [2] CAMELEON <http://giove.cnuce.cnr.it/cameleon.html>
- [3] Cocoon project. <http://xml.apache.org/Cocoon>.
- [4] Vlada Matena and Mark Hapner. Enterprise JavaBeans Specification, v1.1. Technical Report Final Release, Sun Microsystems, Inc., 1999.
- [5] Myers, B., Hudson, S., Pausch, R. *Past, Present, Future of User Interface Tools*. Transactions on Computer-Human Interaction, ACM, 7(1), 2000, pp. 3-28.
- [6] Paternò, F., and Santoro, C. *One Model, Many Interfaces*, Proceedings of CADUI'02, Valenciennes, France, 2002
- [7] UIML <http://www.uiml.org>
- [8] VoiceXML Forum. <http://www.voicexml.org>