

# Adaptation dynamique de l'interaction multimodale dans les environnements ubiquitaires

Slim Ben Hassen  
Orange Labs  
Lannion, France  
slim.benhassen@orange.com

Mariano Belaunde  
Orange Labs  
Lannion, France  
mariano.belaunde@orange.com

## Abstract

The ever-growing interest in ubiquitous environments and mobility calls for the dynamic adaptation of the way users interact with services. This paper introduces an architecture dedicated to the dynamic adaptation of multimodal interactions in the context of ubiquitous environments. Multimodal interaction consists in composing several interaction modalities (e.g. voice and gesture) into a single interaction. The proposed architecture permits the creation of multimodal interactions using modalities provided in the user's environment by third-part devices. Services (e.g. select and watch movies) also provided in the user's environment can be used through these newly created interactions. The proposed solution has been applied to a mobile telecom scenario.

L'intérêt croissant des environnements ubiquitaires et de la mobilité des utilisateurs amène au besoin de l'adaptation dynamique de l'interaction des utilisateurs avec les services. Cet article présente une architecture dédiée à l'adaptation dynamique de l'interaction multimodale dans le contexte des environnements ubiquitaires. L'interaction multimodale est la possibilité d'utiliser plusieurs modalités d'interaction (la voix ou le geste par exemple) pour interagir avec un service. L'architecture proposée permet la création d'interactions multimodales utilisant les modalités présentes dans l'environnement de l'utilisateur. Les services (sélectionner et regarder des films par exemple) peuvent être utilisés par le biais de ces modalités. La solution proposée a été appliquée à un scénario de téléphonie mobile.

KEYWORDS: MDE, Multimodality, Ubiquitous environments

MOTS-CLÉS: IDM, Multimodalité, Environnements ubiquitaires

## 1 Introduction

Aujourd'hui, le succès des services dépend notamment de leur capacité à s'adapter au contexte d'un utilisateur en situation de mobilité.

Actuellement, deux aspects particuliers et importants de cette adaptation sont la possibilité d'exploiter simultanément plusieurs modalités d'interaction (tactile, voix, geste, etc.) et d'utiliser en parallèle plusieurs dispositifs d'entrée/sortie (écrans smartphone, écran TV, microphones, etc.) pour un meilleur confort dans l'utilisation des services.

Néanmoins peu d'outils et de méthodes sont actuellement disponibles pour développer ce type de services exploitant à la fois la multimodalité et la distribution des équipements. Les interfaces multimodales sont un axe des IHMs qui fournit les concepts et les outils pour une interaction homme-machine plus naturelle. Elles permettent à un utilisateur d'interagir avec l'application en entrée et/ou en sortie en utilisant deux ou plus modalités d'interaction.

L'article est organisé comme suit. La section 2 présente le scénario de la transmission de la télévision en direct qui illustre les concepts présentés dans notre travail. Dans la section 3, on explique l'architecture de notre plate-forme, permettant un accès multimodal aux services. La section 4 décrit l'implémentation de notre approche. La section 5 présente les travaux de recherche connexes à l'adaptation des interactions multimodales. La section 6 conclut en synthétisant notre contribution et en donnant les perspectives de notre travail.

## 2 Présentation du cas d'étude

Cette section décrit le scénario de la transmission de la télévision en direct et présente les problématiques auxquelles notre architecture doit répondre.

### 2.1 Scénario de la transmission de la télévision en direct

Un utilisateur, souscrit à un service de transmission de la télévision en direct chez son opérateur de téléphonie mobile, est en voyage dans une ville. Dans sa chambre d'hôtel, il souhaite regarder un programme de télévision. Il remarque qu'un grand écran, un microphone et un dispositif de reconnaissance de geste sont disponibles dans la chambre d'hôtel. Ces dispositifs et les modalités d'interaction qu'elles offrent peuvent être utilisés pour interagir d'une façon multimodale avec le service de transmission de la télévision en direct. Quand l'utilisateur quitte l'hôtel et prend l'avion pour continuer son voyage, il souhaite continuer de regarder le programme de télévision. De nouveaux dispositifs et modalités d'interaction sont disponibles dans son entourage dans l'avion. Ces derniers peuvent être utilisés pour continuer l'interaction avec le service.

### 2.2 Exigences de l'architecture

Une architecture dédiée à un accès multimodal aux services proposés par un opérateur de téléphonie mobile, doit répondre aux problématiques suivantes :

- *Une détection de l'environnement et une découverte des dispositifs d'interaction:* La plate-forme doit être capable de détecter tout changement dans l'environnement de l'utilisateur. Par exemple, elle doit détecter la disparition ou l'apparition d'un dispositif d'interaction. De plus, elle doit être capable de découvrir les caractéristiques des dispositifs d'interaction se situant dans le voisinage de l'utilisateur ainsi que les modalités d'interaction qu'elles peuvent offrir.

- *Sélection des modalités d'interaction:* Après la détection, l'utilisateur peut choisir les modalités d'interaction qu'il souhaite utiliser parmi l'ensemble des modalités offertes par les dispositifs d'interaction présents dans son environnement. Ainsi, la plate-forme doit être capable de fournir une interface utilisateur permettant à l'utilisateur de choisir les modalités d'interaction. Cette interface peut être assimilée à une meta-interface utilisateur.
- *Interaction multimodale:* La plate-forme doit offrir la capacité d'utiliser et de combiner différentes modalités d'interaction en entrée/sortie avec le service, comme le geste et la voix. Elle doit permettre l'utilisation des modalités d'interaction quelque soit la façon, séquentielle ou parallèle.

### 3 Architecture

Cette section présente l'architecture proposée dans cet article.

#### 3.1 Principes fondamentaux de l'architecture

L'architecture qu'on propose est composée de trois principaux blocs logiques.

Le premier bloc, nommé découverte (Discovery), est responsable de la détection et de la découverte des dispositifs et modalités d'interaction dans le voisinage de l'utilisateur.

Le second bloc est représenté par un composant maître nommé contrôleur (Controller). Ce composant gère l'interaction entre l'utilisateur et le service. Cette gestion se fait en instanciant des composants spécifiques d'interaction à l'exécution, selon les modalités d'interaction et leurs combinaisons sélectionnées par l'utilisateur.

Le troisième bloc (Runtime Components) est instancié à l'exécution par le composant contrôleur. Il contient les composants d'interaction comme les composants d'interprétation des modalités d'entrée et le composant de rendu de l'interface utilisateur.

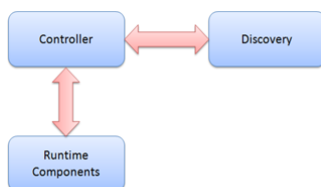


Figure 1: Principaux blocs logiques

#### 3.2 Architecture détaillée

L'architecture présentée dans cet article contient les composants suivants:

- *Interpréteurs des modalités d'entrée pour chaque type de modalité d'interaction:* Un service multimodal reçoit en entrée un flux provenant d'une modalité d'interaction, comme la voix ou le geste, parmi celles offertes par les dispositifs d'interaction présents dans l'environnement de l'utilisateur. Interpréteur d'une modalités d'entrée : 1. reçoit un flux d'entrée depuis un dispositif d'interaction ; 2. détermine l'action à exécuter en

conséquence ; 3. envoie une description de ces actions au composant de fusion et d'intégration.

- *Composant de fusion et d'intégration*: Le composant de fusion et d'intégration permet de fusionner et d'intégrer plusieurs entrées de l'utilisateur pour former une unique interaction pouvant donner lieu à une action du service. Cette interaction est envoyée par la suite au composant de gestion du dialogue.
- *Composant de fission*: Le composant de fission permet de distribuer l'information à présenter à l'utilisateur sur l'ensemble des dispositifs et modalités d'interaction disponibles dans le voisinage de l'utilisateur.
- *Composant de gestion du dialogue*: Ce composant permet de gérer le séquençement des entrées et des sorties avec le service (par exemple l'enchaînement des écrans dans une interface graphique)
- *Composant de rendu de l'interface utilisateur*: Le rôle de ce composant est de fournir l'interface utilisateur finale. De plus, il est responsable de la création de l'interface utilisateur permettant à ce dernier de sélectionner les modalités d'interaction qu'il souhaite utiliser durant son interaction avec le service.
- *Composant de découverte*: Le rôle de ce composant est de détecter et de découvrir les dispositifs et les modalités d'interaction présents dans l'environnement de l'utilisateur.
- *Composant contrôleur*: Ce composant est le cœur de l'architecture présentée dans cet article. Au début, il va instancier le composant de rendu de l'interface utilisateur, qui fournira l'interface utilisateur permettant la sélection d'un ensemble de modalités d'interaction et leurs combinaisons parmi celles découvertes dans l'environnement de l'utilisateur. Ensuite, une fois qu'il a reçu les modèles d'interaction depuis le service et les modalités d'interaction sélectionnées par l'utilisateur, il instanciera à l'exécution un ensemble d'interpréteurs de modalités d'entrée (selon les modalités sélectionnées), le composant de fusion et d'intégration, le composant de fission et le composant de gestion du dialogue et reconfigurera les composants de rendu de l'interface utilisateur. De plus, il a la charge de reconfigurer automatiquement ces composants suite à un changement de contexte.

La figure 2 montre un exemple d'instanciation des composants et des connexions entre eux. Chaque bloc logique présenté dans la section précédente est représenté par un rectangle bleu en pointillé.

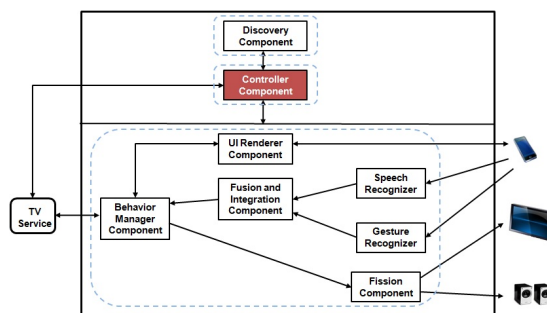


Figure 2: Architecture détaillée

## 4 Implémentation

Dans cette section, nous présentons l'implémentation de la plate-forme présentée dans cet article. Pour implémenter notre contribution, nous avons réutilisé et étendu une solution existante nommée VoiceBench [6]. Ce framework est dédié à la base au développement de services vocaux selon une approche dirigée par les modèles.

La figure 8 donne une vue générale de VoiceBench. Il permet à la fois la création et l'exécution des services vocaux. L'environnement d'exécution est une plate-forme permettant d'exécuter les modèles développés et de déployer le service vocal.

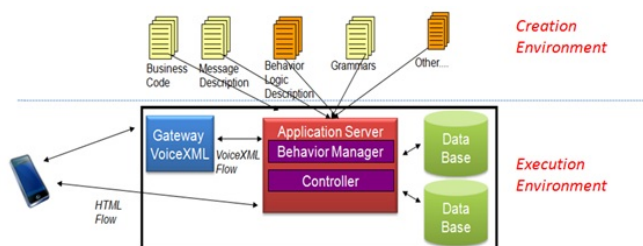


Figure 3: Vue générale de VoiceBench

L'environnement d'exécution de VoiceBench fournit un ensemble de composants nécessaire pour le fonctionnement de notre plate-forme. Comme nous pouvons l'observer sur la figure 8, ces composants sont : le gestionnaire de comportement (Behavior Manager), le contrôleur (Controller), les moteurs de rendu vocal (VoiceXML Flow) et graphique (Html Flow) et l'interpréteur de parole (VoiceXML gateway). Pour couvrir l'ensemble des exigences de notre plate-forme, de nouveaux composants et connexions ont été ajoutés et les composants préexistants ont été modifiés. Par exemple, pour découvrir les dispositifs d'interaction disponibles dans le voisinage de l'utilisateur, un composant implémentant le protocole UPnP [1] a été développé. De plus, ce composant convertit le format des données pour pouvoir produire des instances du métamodèle des dispositifs physiques d'interaction.

## 5 Travaux connexes

Plusieurs applications multimodales ont été développées [5]. Mais, la majorité de ces applications ne permettent d'utiliser que des modalités d'interaction prédéfinies et n'offrent pas la flexibilité nécessaire dans un environnement mobile.

Dans [3], les auteurs présentent une solution permettant un accès multimodal à des documents de maintenance pour des employés en situation de mobilité. Dans l'architecture qu'ils proposent, les tâches demandant des ressources importantes comme la reconnaissance vocale ou gestuelle sont exécutées au niveau du serveur. Ceci permet d'utiliser des dispositifs d'interaction avec des ressources limitées. Le composant de gestion de l'adaptation, coté serveur, transforme une description multimodale abstraite de l'interface utilisateur à des descriptions concrètes de modalités d'interaction et les transmet sur les dispositifs d'interaction des employés de maintenance. Mais, le système développé ne supporte pas l'utilisation des modalités d'interaction en parallèle.

Les travaux décrits dans [2] présentent I-AM (Interaction Abstract Machine), un middleware pour le couplage dynamique des ressources d'interaction dans l'objectif de former un espace d'interaction unifié. L'approche supporte un niveau plate-forme pour manipuler les ressources d'interaction et un niveau logique permettant d'abstraire les ressources d'interaction découvertes dans l'environnement de l'utilisateur. Des interacteurs logiques peuvent par la suite être mappés en interacteurs physiques. Les travaux décrivent une approche supportant la migration d'interfaces utilisateurs et leur distribution dynamique sur un ensemble hétérogène de plate-formes mais ne prend pas en compte la multimodalité (l'approche est limitée à la souris et au clavier en entrée et au graphique en sortie).

Les travaux présentés dans [4] s'intéressent à la collaboration entre PDAs et PCs pour offrir, ce que les auteurs appellent, des interfaces utilisateurs multi-machine. Les composants principaux de l'architecture proposée sont des programmes client s'exécutant sur les PDAs et fournissant l'interface utilisateur, des programmes serveur s'exécutant sur le PC et Pebbles PC, un programme s'exécutant sur le PC et jouant le rôle d'intermédiaire entre les programmes client et les programmes serveur. Sinon, dans la solution proposée dans ces travaux, les ressources d'interaction supportées sont limitées aux PCs et aux PDAs.

## 6 Conclusion

L'adaptation des interactions multimodales au contexte d'un utilisateur en situation de mobilité joue un rôle important pour le succès des services. Dans cet article, nous avons décrit une architecture permettant une interaction multimodale utilisant les modalités disponibles dans l'environnement de l'utilisateur.

Dans nos travaux futurs, des langages spécifiques seront développés pour permettre aux concepteurs de services de spécifier les interactions multimodales et leurs adaptations.

## References

- [1] <http://www.upnp.org/>.
- [2] Nicolas Barralon, Joëlle Coutaz, and Christophe Lachenal. Coupling interaction resources and technical support. In *Proceedings of the 4th international conference on Universal access in human-computer interaction: ambient interaction*, UAHCI'07, pages 13–22, Berlin, Heidelberg, 2007. Springer-Verlag.
- [3] Rodger Burmeister, Christoph Pohl, Siegfried Bublitz, and Pascale Hugues. Snow - a multimodal approach for mobile maintenance applications. In *Proceedings of the 15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*, WETICE '06, pages 131–136, Washington, DC, USA, 2006. IEEE Computer Society.
- [4] Brad A. Myers. Using handhelds and pcs together. *Commun. ACM*, 44(11):34–41, November 2001.
- [5] Sharon Oviatt and Philip Cohen. Perceptual user interfaces: multimodal interfaces that process what comes naturally. *Commun. ACM*, 43(3):45–53, March 2000.
- [6] María José Presso and Mariano Belaunde. Applying mda to voice applications: An experience in building an mda tool chain. In *ECMDA-FA*, pages 1–8, 2005.