Une assistance générique pour les utilisateurs d'applications fortement orientées connaissances

Blandine Ginon, Stéphanie Jean-Daubias, Pierre-Antoine Champin

▶ **To cite this version:**

Blandine Ginon, Stéphanie Jean-Daubias, Pierre-Antoine Champin. Une assistance générique pour les utilisateurs d'applications fortement orientées connaissances. 23es Journées Francophones d'Ingénierie des Connaissances - IC 2012, Jun 2012, Paris, France. pp.167-174. hal-00816683

HAL Id: hal-00816683
https://hal.science/hal-00816683

Submitted on 22 Apr 2013

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

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

Une assistance générique pour les utilisateurs d'applications fortement orientées connaissances

Blandine Ginon[1,2], Stéphanie Jean-Daubias[1,3] et Pierre-Antoine Champin[1,3]

[1] Université de Lyon, CNRS,
[2] INSA-Lyon, LIRIS, UMR5205, F-69621, France
[3] Université Lyon 1, LIRIS, UMR5205, F-69622, France
{prenom.nom}@liris.cnrs.fr

Résumé : Cet article présente un modèle d'assistant générique pour les utilisateurs d'applications informatiques, en particulier les applications fortement orientées connaissances. Ce modèle s'appuie sur deux formalismes : un formalisme permettant de décrire un assistant et la manière de le paramétrer, et un formalisme permettant de décrire l'assistance souhaitée pour une application donnée. Cette description sera ensuite exploitée par un assistant générique qui fera appel à des assistants épiphytes pour proposer à chaque utilisateur une assistance entièrement personnalisée en fonction de ses besoins.

Mots-clés : Assistance à l'utilisateur, modèles, formalismes de description.

1 Introduction

Ces dernières années, de nombreuses applications fortement orientées connaissances ont été développées. Ces applications sont conçues pour collecter, capitaliser et surtout réutiliser des connaissances issues des expériences humaines (Cordier, et al., 2010). Elles se caractérisent par des modèles souvent génériques sur lesquels elles s'appuient et par les très riches possibilités qu'elles offrent aux utilisateurs. Néanmoins, devant les nombreuses fonctionnalités offertes par ces applications, et leur interface souvent chargée, un utilisateur novice peut renoncer à les utiliser ou sous-exploiter ces applications (Cordier, et al., 2010), limitant ainsi leur intérêt et leur richesse. Pour pallier cette difficulté de prise en main et d'utilisation, il existe deux pistes principales de travail non exclusives : l'amélioration de l'ergonomie et l'assistance à l'utilisateur de ces systèmes. Tout d'abord, la mise en place d'une bonne ergonomie dans une application permet d'augmenter nettement son utilisabilité. Pour cette raison, la qualité de l'ergonomie doit être une importante

préoccupation des concepteurs d'applications, notamment pour les applications fortement orientées connaissances, souvent intrinsèquement complexes. Néanmoins, cela ne permet pas de régler tous les problèmes d'utilisation. Pour cette raison, une fois écartés les problèmes d'utilisation liés à l'ergonomie d'une application, nous avons choisi d'orienter nos recherches vers l'assistance à l'utilisateur.

L'assistance à l'utilisateur englobe tout ce qui peut permettre à utilisateur d'éviter de sous-exploiter une application, ou de se tourner vers un autre utilisateur, plus expert, pour l'aider à surmonter ses difficultés. Elle s'adresse à tous les utilisateurs qu'ils soient novices, occasionnels, standards ou experts. L'assistance concerne aussi bien la prise en main que l'utilisation courante d'une application.

L'enjeu de notre travail est de proposer un modèle d'assistant générique capable de s'adapter à une application existante pour fournir une assistance pertinente à chaque utilisateur. Le dispositif conçu devra proposer différentes approches d'assistance pouvant être combinées, afin de répondre le plus efficacement possible aux besoins des utilisateurs. Pour cela, l'assistance devra prendre en compte les spécificités de l'application et être adaptée au contexte dans lequel se situe le couple utilisateur-application. Enfin, l'assistance proposée devra être entièrement personnalisée en fonction de l'utilisateur, de ses connaissances, capacités, préférences, objectifs, expériences…

Cet article vise à présenter notre modèle générique d'assistance à l'utilisateur, conçu pour être particulièrement adapté aux applications fortement orientées connaissances.

2 Assistance à l'utilisateur

Cette section donne un état de l'art du domaine l'assistance à l'utilisateur. Nous présentons tout d'abord des travaux concernant la modélisation de l'assistance et de l'utilisateur, avant de nous intéresser à l'approche des systèmes épiphytes.

Modèle de l'assistance. Dans le but de proposer un modèle générique d'assistance à l'utilisateur, nous nous intéressons à la modélisation de l'assistance, et plus particulièrement aux modèles génériques, indépendants de l'application dans laquelle ils seront mis en œuvre. Nous avons identifié deux travaux adoptant cette approche.

La première approche est un modèle de système conseiller générique. Un système conseiller est un système informatique qui produit des conseils à partir des interactions entre un utilisateur et une application (Paquette, et al., 1996). Dans ce cadre, un conseil est une indication pour l'utilisateur affichée à sa demande ou sur décision du système conseiller. Le modèle générique de système conseiller proposé par (Paquette, et al.,

1996) permet de concevoir des systèmes conseillers pouvant être greffés sur une application existante. Cette approche a été mise en œuvre dans le système Epitalk, qui permet au concepteur de l'assistance de définir un système conseiller adapté à une application donnée. Toutefois, les applications concernées doivent obligatoirement être implémentées en Smalltalk, et l'utilisation d'Epitalk nécessite une bonne connaissance de la programmation en Smalltalk et de la programmation multi-agents, ce qui limite fortement sa portée.

La seconde approche est le modèle d'assistance générique proposé par (Dufresne, et al., 2003), destiné aux acteurs du téléapprentissage. Il est constitué d'un modèle de la tâche, d'un modèle de l'usager et du groupe, ainsi que d'un ensemble d'interventions d'assistance. Une intervention d'assistance permet de répondre aux besoins des utilisateurs en leur fournissant une assistance adaptée, elle est associée à un ensemble de métadonnées (identifiant, but…) et à un ensemble de règles du type « Si Condition d'assistance, Alors Action d'assistance ». Ce modèle d'assistant générique peut être adapté pour les utilisateurs de logiciels fortement orientés connaissances. Néanmoins, pour que ce modèle soit réutilisé dans le cadre de nos travaux, les actions d'assistance proposées (adaptation de l'interface, affichage de messages, modification du modèle utilisateur…) devront être enrichies pour répondre aux besoins les plus variés des utilisateurs de ce type d'applications.

Modèle de l'utilisateur. Pour permettre une personnalisation de l'assistance, il est nécessaire de posséder des informations sur les utilisateurs. Ces informations peuvent concerner les connaissances d'un utilisateur, comme son niveau de maîtrise de l'application ou plus globalement son niveau de maîtrise de la lecture, sur ses capacités et éventuels handicaps, comme un taux d'audition ou un type de daltonisme, ainsi que ses préférences, comme ses couleurs et polices favorites ou le mode d'assistance qu'il préfère.

Pour pouvoir exploiter ces informations concernant un utilisateur, il est nécessaire qu'elles soient exprimées selon un même formalisme, à travers un profil d'utilisateur. Un profil d'utilisateur est un ensemble de données informatiques concernant un utilisateur. Ces informations peuvent être collectées ou déduites à partir de séances d'utilisation d'un logiciel, ou renseignée directement par l'utilisateur.

Nous avons choisi d'utiliser le langage de modélisation de profils PMDLe (Eyssautier-Bavay, et al., 2011) (Ginon, et al., 2011). Il permet de décrire la structure d'un profil, c'est-à-dire l'organisation des informations contenues dans le profil. Pour cela, PMDLe distingue dans un profil d'utilisateur sa partie structure, pouvant être commune à plusieurs profils, par exemple à tous les utilisateurs d'une même application, de sa partie données, contenant les informations propres à chaque utilisateur. Le langage de modélisation de profils PMDLe est

entièrement mis en œuvre dans EPROFILEA (Jean-Daubias, et al., 2009), qui permet de définir et d'exploiter des profils.

Personnalisation de l'assistance. Pour qu'une assistance soit à la fois efficace et acceptée par l'utilisateur, il est intéressant de la personnaliser, d'une part en fonction de l'application et du contexte dans lequel se trouve l'utilisateur, et d'autre part en fonction des spécificités de l'utilisateur lui-même. Ainsi, on ne proposera pas le même type d'assistance à un utilisateur novice et à un utilisateur expert, et l'assistance proposée sera différente dans le cadre d'une fonctionnalité basique de l'application et dans le cadre d'une fonctionnalité avancée. De même, on ne proposera pas la même assistance à un utilisateur qui apprécie une assistance proactive, qui lui offre des conseils lorsque le système le juge opportun, et à un utilisateur qui préfère une assistance réactive qu'il sollicitera lui-même si besoin et qui ne l'interrompra pas dans sa tâche.

Une approche pour permettre la prise en compte des spécificités des utilisateurs (connaissances, capacités, préférences…) consiste à définir des contraintes portant sur le profil de l'utilisateur.

Le langage de modélisation de profils PMDLe est associé à un modèle de contraintes sur profils, cPMDLe (Lefevre, 2009) (Ginon, et al., 2011), qui permet de contraindre des profils PMDLe. Actuellement, cPMDLe est mis en œuvre dans l'environnement EPROFILEA (Lefevre, et al., 2011) (Ginon, et al., 2011), pour la personnalisation d'activités pédagogiques. Utilisé dans un autre contexte que celui des EIAH, cPMDLe permet également de contraindre des profils d'utilisateurs décrits à l'aide de PMDLe. Ainsi, cPMDLe permet par exemple de sélectionner les utilisateurs dont le niveau de connaissance de l'application est « novice », ou ceux dont l'audition est comprise entre 60% et 80%. Les contraintes sur des profils d'utilisateurs PMDLe définies à l'aide de cPMDLe rendent possible une personnalisation de l'assistance en fonction des spécificités de chaque utilisateur exprimées dans son profil : connaissances, compétences, capacités, préférences, besoins, objectifs…

Systèmes épiphytes. Un système épiphyte est un système qui se greffe sur un autre système, appelé système hôte, sans en gêner le fonctionnement (Paquette, et al., 1996). Dans le contexte de l'assistance à l'utilisateur, un système assistant peut être intégré de manière épiphyte à une application, afin de fournir une assistance aux utilisateurs de cette application. Ainsi, le système assistant est indépendant de l'application, et ne nécessite pas de travail de redéveloppement de cette application.

Plusieurs travaux existants proposent une assistance épiphyte. C'est le cas du système conseiller proposé par (Kosba, et al., 2005), qui se greffe de manière épiphyte à un site web d'enseignement à distance pour fournir des conseils aux enseignants. De même, le système conseiller de (Richard, et al., 2004) se greffe de manière épiphyte à un site web

complexe pour aider les utilisateurs dans la recherche d'information. Cependant, ces systèmes conseillers ne sont pas génériques, ils ne peuvent être appliqués qu'aux sites web auxquels ils sont destinés.

Certains travaux proposent néanmoins une assistance tout à la fois épiphyte et générique. Par exemple, (Microsoft, 2012) propose un ensemble de compagnons pouvant être greffés de manière épiphyte sur de nombreuses applications qui ne se limitent pas aux applications développées par cet éditeur de logiciels. Ces compagnons sont capables de s'exprimer textuellement et oralement pour assister les utilisateurs. De même, (Martin, et al., 2012) proposent un ensemble d'agents conversationnels animés adaptables de manière épiphyte aux applications web. Ces agents s'expriment textuellement et peuvent répondre à des questions en langage naturel. Tout comme les compagnons de Microsoft, ils peuvent également appuyer leur discours par des gestes précis, pointer du doigt un élément de l'interface par exemple. Par ailleurs, le système Cantoche propose des outils permettant de créer des agents conversationnels animés pour une application donnée, ainsi que des foires aux questions, des vidéos de démonstrations et des tutoriels mettant en scène ces agents (Cantoche, 2012). Néanmoins, ces travaux génériques et épiphytes ne proposent chacun qu'une seule approche d'assistance, il serait donc intéressant de pouvoir combiner ces approches afin de répondre au mieux aux besoins des utilisateurs.

3 Notre approche d'assistance à l'utilisateur

Dans cette section, nous présentons notre modèle d'assistant générique pour les utilisateurs d'applications informatiques. Le principe de notre approche est de permettre dans un premier temps au concepteur de l'assistance d'une application de décrire l'assistance souhaitée, et de mettre en place cette assistance en fournissant dans un second temps une assistance personnalisée à chaque utilisateur de cette application, conformément à la description de l'assistance spécifiée par le concepteur.

Notre approche fait appel à un ensemble d'assistants (cf. Ⓐ **FIG. 1**), adaptables de manière épiphyte à différentes applications-cibles. Il peut s'agir par exemple de systèmes conseillers ou d'agents conversationnels. Par la suite, de nouveaux assistants pourront être ajoutés à cet ensemble. Notre approche permet d'adapter et combiner ces assistants pour fournir l'assistance souhaitée par le concepteur, pour une application donnée. Pour permettre l'utilisation de ces assistants épiphytes, ces derniers doivent être décrits à l'aide d'un formalisme commun : un langage de description d'assistants (cf. Ⓑ **FIG. 1**). Cette description permettra par la suite à notre assistant générique de piloter ces assistants et de les greffer sur l'application-cible. Cette description sera faite une seule fois par un expert lors de la création d'un nouvel assistant.

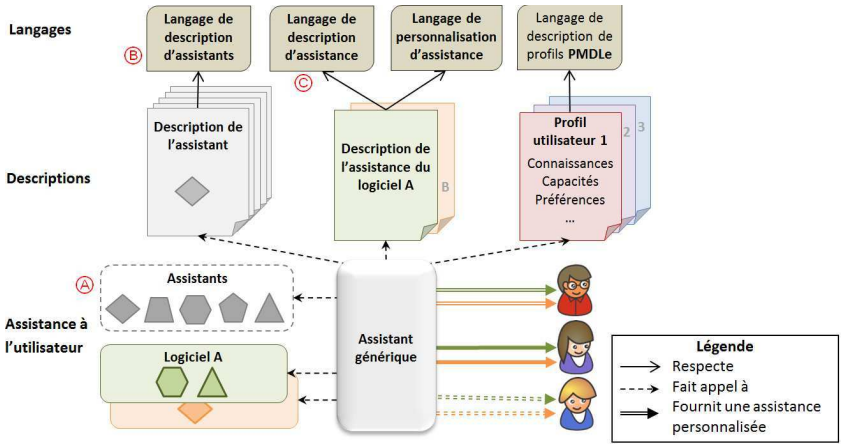

FIG. 1 – *Modèle d'assistant générique.*

Afin de mettre en place l'assistance aux utilisateurs d'une application donnée, le concepteur doit décrire l'assistance souhaitée. Afin que cette description soit ensuite exploitée par l'assistant générique pour assister un utilisateur, elle doit respecter un formalisme de description d'assistance, cf. Ⓒ **FIG. 1**.

Prenons l'exemple d'une application disposant d'un système conseiller intégré et à laquelle un concepteur souhaite ajouter un système d'assistance plus complet, combinant plusieurs techniques d'assistance et capable de fournir à chaque utilisateur une assistance personnalisée. Dans la description de l'assistance souhaitée, le concepteur pourra par exemple spécifier qu'un tutoriel et un agent conversationnel devront être ajoutés à l'application. Il pourra également spécifier que le tutoriel sera conseillé aux utilisateurs novices et disponible sur demande pour les autres utilisateurs. Le choix de l'avatar de l'agent conversationnel pourra dépendre des préférences de l'utilisateur. Quant au système conseiller qui est intégré à l'application, le concepteur pourra le personnaliser en spécifiant que le système conseiller interviendra fréquemment pour donner des conseils simples aux utilisateurs novices, alors qu'il interviendra peu auprès des utilisateurs experts, mais en privilégiant les conseils très précis et techniques liés à la tâche en cours.

La description de l'assistance d'une application spécifie d'une part les techniques d'assistance que le concepteur souhaite mettre en œuvre dans l'assistance. Elle spécifie d'autre part la personnalisation souhaitée par le concepteur. Pour cela, le concepteur crée des règles d'assistance de type « Si Condition d'assistance, Alors Action d'assistance ». Dans la spécification des conditions d'assistance, le concepteur peut combiner des conditions portant sur le profil de l'utilisateur (telle que « Si l'utilisateur est novice »), avec des conditions portant sur le contexte (telle que « Si l'utilisateur a cliqué sur le bouton Y »). Dans les actions d'assistance, le concepteur précise quel assistant va réaliser l'action :

cette action doit donc respecter la description de l'assistant, c'est-à-dire être une action possible de cet assistant. Par exemple, si l'assistant est un agent conversationnel animé dont la description précise qu'une action doit contenir un message et un ensemble d'animations (parmi une liste d'animations possibles), alors une action d'assistance pourrait être « message = « Passez à l'étape suivante » ; animation = « pointer le bouton Suivant » ; animation = « sourire » ». Une fois que le concepteur de l'assistance d'une application aura spécifié la description de l'assistance souhaitée, notre assistant générique fournira à chaque utilisateur l'assistance souhaitée. Pour cela, l'assistant générique fait appel à la description de l'assistance du logiciel, ainsi qu'aux descriptions des assistants impliqués et au profil de l'utilisateur. L'interaction avec l'application permet à l'assistant générique de piloter les assistants épiphytes lorsque leur action est requise, mais également de personnaliser le système d'assistance éventuellement intégré à l'application pour que l'assistance qu'il fournit respecte la description de l'assistance.

Notre modèle permet donc au concepteur de l'assistance d'une application de définir l'assistance souhaitée, ainsi que la personnalisation de cette assistance. À l'aide de cette description, notre assistant générique pourra par la suite fournir à chaque utilisateur de l'application une assistance entièrement personnalisée et respectant les choix du concepteur de l'assistance.

4 Conclusion et perspectives

Face au développement récent d'applications fortement orientées connaissances, un besoin d'assister les utilisateurs de telles applications a émergé, en raison des larges possibilités offertes par ces applications, et de leur interface souvent chargée.

Dans cet article, nous avons présenté notre modèle d'assistance générique à l'utilisateur, particulièrement adapté aux applications fortement orientées connaissances. Il permet de proposer plusieurs types d'aides pour une application donnée, et de personnaliser l'aide fournie à un utilisateur. En effet, les fonctionnalités des applications fortement orientées connaissances étant variées et complexes, un seul type d'aide ne peut pas convenir à toutes les tâches ni à tous les utilisateurs. Ce modèle se base sur un formalisme de description de l'assistance, qui permettra à un concepteur d'aide de définir l'assistance souhaitée pour une application donnée. Cette description sera ensuite utilisée par un assistant générique qui fournira à chaque utilisateur de l'application l'assistance spécifiée, de manière entièrement personnalisée en fonction des choix du concepteur. Pour cela, l'assistant générique pourra faire appel à différents assistants épiphytes.

Dans la continuité de ce travail de recherche, nous souhaitons désormais définir le formalisme de description d'assistance, qui permettra la spécification d'assistances riches et personnalisables, adaptées aux applications les plus variées. Nous souhaitons également définir le formalisme de description d'assistants, afin de rendre possible la réutilisation et la personnalisation de ces assistants. Nous mettrons ensuite en œuvre ces formalismes dans deux logiciels destinés aux concepteurs d'assistance.

Références

CORDIER, A., LEFEVRE, M., JEAN-DAUBIAS, S. & GUIN, N. (2010). Concevoir des assistants intelligents pour des applications fortement orientées connaissances : problématiques, enjeux et étude de cas. In IC 2010 - 21èmes Journées Francophones d'Ingénierie des Connaissances, p 119-130.

PAQUETTE, G., PACHET, F., GIROUX, S. & GIRARD, J. (1996). EpiTalk, a generic tool for the development of advisor systems. p 349-370.

DUFRESNE, A., BASQUE, J., PAQUETTE, G., LÉONARD, M., LUNDGREN-CAYROL, K. & PROMTEP, S. (2003). Vers un modèle conceptuel générique de système d'assistance pour le téléapprentissage. p 57-88.

EYSSAUTIER-BAVAY, C. & JEAN-DAUBIAS, S. (2011). PMDL: a modeling language to harmonize heterogeneous learners profiles. In ED-MEDIA 2011, Lisbon Portugal.

GINON, B., JEAN-DAUBIAS, S. & LEFEVRE, M. (2011). Evolutive learners profiles. In ED-MEDIA 2011, Lisbon Portugal.

JEAN-DAUBIAS, S., EYSSAUTIER-BAVAY, C. & LEFEVRE, M. (2009). Modèles et outils pour rendre possible la réutilisation informatique de profils d'apprenants hétérogènes. Sticef 2009, vol 12, p 171-208.

LEFEVRE, M. (2009). Processus unifié pour la personnalisation des activités pédagogiques : méta-modèle, modèles et outils. Thèses de doctorat, université Lyon 1.

GINON, B. & JEAN-DAUBIAS, S. (2011). Models and tools to personalize activities on learners profiles. In ED-MEDIA 2011, Lisbon Portugal.

LEFEVRE, M., JEAN-DAUBIAS, S. & GUIN, N. (2011). Adapte, un logiciel pour aider l'enseignant à proposer des activités personnalisées à chacun de ses apprenants. In EIAH'2011 - Atelier "Personnalisation de l'apprentissage : quelles approches pour quels besoins ?".

KOSBA, E., DIMITROVA, V. & BOYLE, R. (2005). Using student and group models to support teachers in web-based distance education. In User Modeling 2005, p 124-133, Edinburgh, Scotland, UK.

RICHARD, B. & TCHOUNIKINE, P. (2004). Une approche centrée modèle pour la construction d'un système conseiller pour un site web. In IC 2004, p 151-162, Lyon, France.

MICROSOFT. (2012). http://www.microsoft.com/products/msagent/main.aspx.

MARTIN, J.-C. AND SANSONNET, J.-P. (2012). http://perso.limsi.fr/jps/online/weblea/leaexamples/leawebsite/index.html.

CANTOCHE. (2012). http://www.cantoche.com/fr~LivingActor.html.