



Groupe de travail CESAME : Compte rendu de la réunion des 3, 4 et 5 juin 2006 à Sophia-Antipolis

1. Participants

Grenoble	Gaëlle Calvary, Sophie Dupuy, Benoit Mansoux
Lyon	Franck Tarpin-Bernard
Sophia-Antipolis	Mireille Blay Fornarino, Anne-Marie Dery, Stéphane Lavirotte, Oana Novacescu, Philippe Renevier, Jean-Yves Tigli
Toulouse	Emmanuel Dubois, Guillaume Gauffre, Jean-Pierre Jessel
Valenciennes	Anas Hariri

2. Agenda

Journée du 3 mai

- 10H30 Accueil
- 11H00 Rappel des objectifs du groupe et résumé de la première réunion.
Présentation des journées.
Rappel du principe des présentations : exposés courts (15 minutes) de positionnement par rapport à l'étude de cas CESAME (état des lieux).
- < Repas >
- 14H00-16h00 : Ingénierie Dirigée par les Modèles (IDM)
Gaëlle Calvary : IDM&IHM : pour le meilleur et pour le ...
Guillaume Gauffre : Couplage de Modèles de Conception pour les Systèmes Mixtes: quelle place pour l'IDM ?
- 16h30-17h30 Systèmes mixtes
Sophie Dupuy : Vers une méthode de conception pour les systèmes mixtes.
Benoit Mansoux : Interaction multimodale en sortie pour les systèmes mixtes.
- 17H30- 19H00 Composants
Oana Novacescu, Anne-Marie Dery, Philippe Renevier : pbm : peut on déduire la composition d'IHM de la composition des objets métiers ?
Stéphane Lavirotte, JY Tigli: présentation de l'ubiquarium

Journée du 4 mai

- 9H30-12h00 : Plasticité
Gaëlle Calvary : les comets : des widgets pour la plasticité
Anas M. Hariri : Principes de base d'une démarche de génération d'IHM plastiques
Franck Tarpin-Bernard : Et si on recyclait AMF pour la plasticité ?
- <Repas>
- 13h30-18h00 Positionnement comparatif des travaux
- 19h00 Dîner

Journée du 5 mai

- 9H30-12h00 : Synthèse et perspectives
Suite des discussions puis synthèse et préparation du futur pour le groupe.

3. Résumé des présentations

Les résumés sont fournis par les auteurs. Ils sont donnés dans l'ordre des présentations (agenda).

3.1 Gaëlle Calvary - IDM&IHM : pour le meilleur et pour le ...

Travail de Jean-Sébastien Sottet (Thèse 1^{ère} année), co-encadré par Gaëlle Calvary (IHM) et Jean-Marie Favre (IDM).

Description

La communauté IHM fait des modèles et les transforme depuis toujours. En Génie Logiciel, une communauté s'élève : l'Ingénierie Dirigée par les Modèles (IDM). Le tandem IHM&IDM se soude pour l'ingénierie d'IHM avancées. Un système interactif devient une toile de modèles, support à la capture et l'expression de son design rationale et de son évolution à l'exécution, en particulier de sa plasticité. Tels des élastiques, les liens entre modèles (mappings) relient les modèles avec l'ambition de respecter le cahier des charges. La tension des élastiques et éventuellement leur rupture imagent la limite de plasticité. Un pas vers l'octopus doué, comme nos futurs systèmes interactifs, de remodelage et redistribution ...

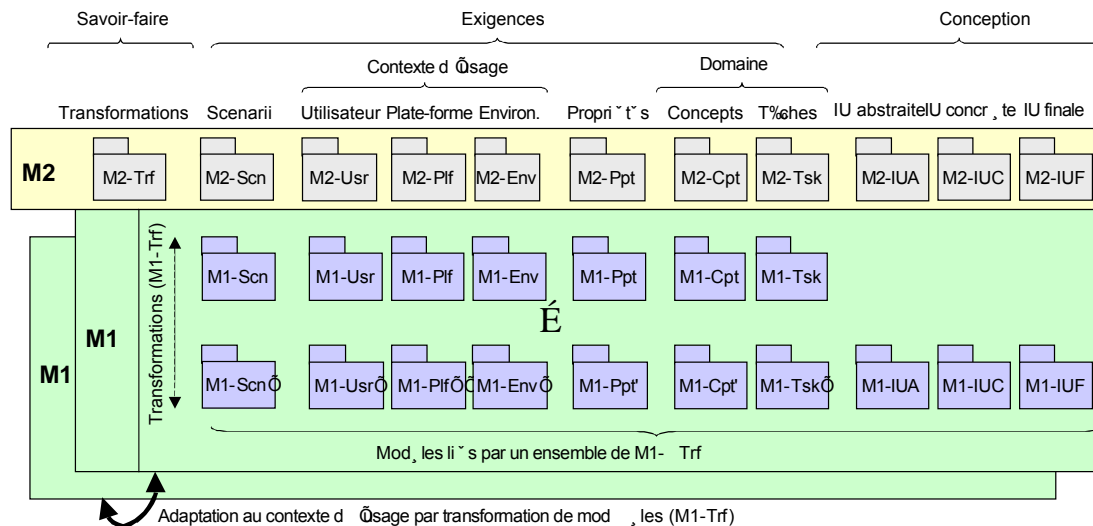


Schéma mis à jour suite aux remarques du groupe. MERCI !

Références

Sottet, J.S., Calvary, G., Favre, J.M., Coutaz, J., Demeure, A., Balme, L. Towards Model-Driven Engineering of Plastic User Interfaces, In the ACM/IEEE 8th International Conference on Model Driven Engineering Languages and Systems (MODELS'05) satellite proceedings, Springer LNCS, pp 191-200.

Sottet, J.S., Calvary, G., Favre, J.M., Coutaz, J. IHM&IDM : Un tandem prometteur, Article court à ERGO'IA 2006, 10ème conférence ERGO-IA, L'humain comme facteur de performance des systèmes complexes, 11-13 Octobre 2006, Biarritz, France, pp 79-86

3.2 Guillaume Gauffre - Couplage de Modèles de Conception pour les Systèmes Mixtes: quelle place pour l'IDM ?

Travail de Guillaume Gauffre (M2R), co-encadré par Emmanuel Dubois et Rémi Bastide.

Description

Le modèle ASUR permet de décrire les différentes facettes de l'interaction d'un utilisateur avec un système mixte au cours de la réalisation d'une tâche. Il s'appuie pour cela sur l'identification des différents composants physiques et numériques impliqués dans une interaction mixte et l'identification des échanges de données entre ces composants. Support à l'exploration des techniques d'interaction possible, à leur description et à leur comparaison, le modèle ASUR n'offre toutefois aucune aide directe pour l'implémentation de ces systèmes.

Afin d'y remédier, nous proposons l'extension ASUR-IL (Implementation Layer) pour décrire la structure logicielle du système mixte à développer. Sur la base des concepts du domaine et échanges identifiés dans la modélisation ASUR, les diagrammes ASUR-IL décrivent les composants logiciels et dispositifs entrant en jeu dans l'interaction avec l'utilisateur. Cet assemblage de composants est composé d'entités ASUR-IL, d'adaptateurs ASUR-IL et d'échanges de données. A travers la structure des entités et leurs connexions avec les adaptateurs, interaction du monde physique vers le monde numérique (interaction en entrée) et celle du monde numérique vers le monde physique (interaction en sortie) sont clairement

distinguées. Chaque diagramme ASUR-IL peut ensuite donner lieu à une implémentation via le modèle à composants de la plateforme WComp : chaque élément ASUR-IL est alors associé à un composant WComp.

ASUR-IL s'inscrit dans la continuité de la conception faite avec le modèle ASUR, en exploitant et complétant les composants, relations et caractéristiques mis en évidence en phase de conception. ASUR-IL permet donc d'exprimer un premier lien entre un modèle de conception de haut niveau et une description implémentable du système. De plus, via la définition précise du découpage logiciel adapté à l'implémentation du système modélisé, ASUR-IL permet aussi d'effectuer le lien entre le modèle ASUR et une approche à composants, ici WComp.

Nos travaux s'orientent dorénavant vers l'utilisation de l'Ingénierie Dirigée par les Modèles afin d'exprimer ces deux liens de manière formelle. Ainsi, nous obtiendrons un support à l'articulation des différentes étapes de conception des systèmes mixtes. Nous pourrions par la suite envisager la prise en compte d'autres éléments de conception via l'utilisation d'autres modèles : modèles de contextes, de tâches, de plateformes, ..., ou bien d'autres modèles de composants.

Références :

- Dubois, E., Gauffre, G., Bach, C., and Salembier, P., "Participatory Design Meets Mixed Reality Design Models - Implementation based on a Formal Instrumentation of an Informal Design Approach", *Chapter 6, in G. Calvary, C. Pribeanu, G. Santucci, J. Vanderdonck (eds.), "Computer-Aided Design of User Interfaces V", Proc. of 6th Int. Conf. on Computer-Aided Design of User Interfaces CADUI'2006 (Bucharest, 6-8 June 2006), Information Systems Series, Springer-Verlag, Berlin, 2006, pp. 71-84.*

- Gauffre, G., *Master2 2IH report : Techniques d'interaction augmentée : vers un lien entre modélisation et conception logicielle*, Université Paul Sabatier Toulouse 3, June 2005

3.3 Sophie Dupuy - "vers une méthode de conception pour les systèmes mixtes"

Travail de David Juras (CNAM) encadré par Sophie Dupuy et Dominique Rieu.

Description

Nous proposons d'étendre une méthode de développement logiciel en synchronisant les activités des différents acteurs (ingénieur logiciel, spécialiste interaction homme-machine, ergonomiste) afin de proposer un processus couvrant à la fois la conception du noyau fonctionnel et de l'interaction homme-machine, en particulier dans le cas des systèmes mixtes. Notre méthode est formalisée sous forme de patrons et est outillée dans l'atelier de gestion de patrons AGAP qui permet d'obtenir un guide méthodologique sous forme d'un site web. En terme de modèles, la méthode se base sur le standard de modélisation en génie logiciel UML et pour la partie IHM sur les arbres de tâches et ASUR.

Références

D. Juras, D. Rieu, S. Dupuy-Chessa, Agnès Front, Conception collaborative pour les systèmes mixtes, Inforsid, 2006, Hammameth, Tunisie, Juin 2006 à paraître

D. Juras, D. Rieu, S. Dupuy-Chessa, Vers une méthode de développement pour les systèmes mixtes, Journée NEPTUNE 2006, Revue Génie Logiciel, juin 2006, à paraître.

3.4 Benoit Mansoux - Interaction multimodale en sortie pour les systèmes mixtes.

Travail de Benoit Mansoux (thèse 3^{ème} année), encadré par Laurence Nigay et Jocelyne Trocaz.

Description

ICARE est une approche à composants utilisée ici pour la conception et le prototypage rapide d'interfaces multimodales en sortie (du système vers l'utilisateur). Les modalités représentées par des composants ICARE élémentaires (dispositifs et langages d'interaction) peuvent être combinées entre elles grâce à des composants ICARE génériques, issus des propriétés CARE pour la multimodalité.

Les applications visées sont des applications de Chirurgie Assistée par Ordinateur, où l'on trouve généralement des entités physiques et numériques participant à l'interaction.

A la différence d'autres approches comme WComp, ICARE n'est pas dynamique, les schémas de composition des composants ne peuvent pas être modifiés au run-time.

ICARE est implémenté suivant le modèle à composants JavaBeans, la communication entre composants est assurée par un mécanisme d'événements.

Références

J. Bouchet, L. Nigay*

ICARE : Approche à composants pour l'interaction multimodale

Actes des Premières Journées Francophones : Mobilité et Ubiquité 2004, Cépadués-Éditions. Nice Sophia-Antipolis, France.

B. Mansoux, L. Nigay*

Distributed Display Environments in Computer-Assisted Surgery systems

CHI05 Workshop: The Future of User Interface Design Tools, Portland, Oregon, USA.

Mansoux, B., Nigay, L. & Troccaz, J. [2006], Output Multimodal Interaction: The Case of Augmented Surgery, Proceedings of BCS-HCI 2006 conference, Septembre 2006, (to appear)

* : Publications disponibles sur <http://iihm.imag.fr/>

3.5 Oana Novacescu - Peut on déduire la composition d'IHM de la composition des objets métiers ?

Travail de Oana Novacescu (M2R), encadrée par Anne-Marie Dery et Philippe Renevier

Description

Notre travail se base sur un modèle à composant qui associe à un composant métier cié un ou plusieurs composants IHM. Ce dernier est décrit de manière abstraite, pour garantir son indépendance par rapport au support d'exécution sous-jacent. Nous considérons actuellement une interface abstraite comme une structure de données arborescente (toujours abstraite) de style XML. Nous nous reposons sur l'existant (UsiXML [1] notamment) pour la définition et la « concrétisation » des interfaces abstraites. Ces abstractions vont être utilisées comme un métaobjet (cf. MOP méta object protocol) qui va être lié à un composant métier.

Dans notre travail, nous avons dégagé plusieurs étapes : la première que l'on peut qualifier de plastique consiste à essayer de déterminer la « bonne interactivité ». Dans ce contexte nous étudions l'adaptation d'une application écrite sous forme d'assemblage de composants dans un certain contexte d'utilisation.

Pour atteindre cet objectif, nous considérons que les interactions entre composants sont différentes et doivent être modifiées en fonction du type d'utilisateur. Ces interactions entre les composants métiers seuls et entre les composants métiers et les composants IHM sont décrites dans le langage ISL (Interaction Specification Language [2]). Une interaction représente en quelque sorte la facette contrôle d'un agent PAC ou AMF. Dans ce contexte, notre proposition se base sur le *service* d'interactions NOAH [3]. NOAH permet la définition et la modification d'interactions entre composants de manière dynamique ou persistante. Selon le contexte d'exécution d'une interaction entre deux composants métiers, un composant d'IHM abstrait d'un des composants métiers est choisi et assemblé avec la partie IHM du second composant métier.

La deuxième étape consiste en une déduction systématique de ce que nous appelons une « bonne » *combinaison* de composants IHM. Cette bonne combinaison est réalisée en fonction des interactions existantes entre composants métier seuls, et entre des composants métier et leurs composants IHM.

Les premiers résultats sont illustrés par des exemples d'application dans la plate-forme Amusing [2]. Elle prend pour entrées plusieurs descriptions d'IHM spécifiées en SunML [2] et produit en sortie une *combinaison* (soit l'union, l'intersection, la différence et quelques autres opérateurs) de ces descriptions. Amusing permet l'assemblage d'interfaces à partir des opérateurs définis ci-dessus et spécifiés pour l'occasion par l'utilisateur. Ainsi la construction et la déduction de nouvelle interface ne sont pas automatiques. Nous pensons qu'en nous basant sur les interactions entre les composants métiers nous pourrions alors déduire des règles d'assemblage pour les IHM.

Références

[1] Limbourg, Q., Vanderdonkt, J., Michotte, B., Bouillon, L., Florins, M., Trevisan, D. USIXML: A User Interface Description Language for Context-Sensitive User Interfaces, Workshop on Device Independent Web Engineering DIWE'04 (Munich, Germany, July 26-27), M. Lauff (Ed.), 2004.

[2] Anne-Marie Pinna-Déry, Jérémy Fierstone, Michel Riveill, E. Picard. "User Interface: a Technical Component in Component-Based Models in Response to Human Computer Interaction Adaptation" Research Report I3S, number RR-2004-06-FR, 24 pages, Sophia Antipolis (France), feb 2004. disponible à l'adresse suivante :

<http://www.i3s.unice.fr/%7Emh/RR/2004/RR-04.06-AM.DERY.pdf>

[3] Blay-Fornarino, M., Charfi, A., Emsellem, D., Pinna-Déry A.M., Riveill, M. Software interaction. Journal of Object Technology, ETH Zurich, 2004, 161-180.

3.6 J.-Y. Tigli, S. Lavirotte, D. Cheung, V. Hourdin - WComp et Ubiquarium

Description

L'avènement de nouveaux usages de l'informatique dite mobile et ambiante ne permet plus de concevoir des applications logicielles dédiées à des plates-formes prédéfinies, standardisées, composées d'un ensemble de dispositifs connus a priori. Les logiciels nécessitent toujours plus de capacité d'adaptation face à une multitude de contextes d'utilisation. Que dire alors de l'enjeu proposé par une informatique qui se voudrait adapter dynamiquement une application logicielle à un environnement d'exécution découvert dynamiquement, évoluant tout aussi dynamiquement, et partiellement connue a priori. Le paradigme qui permet de gérer une telle application par assemblage de composants se révèle alors particulièrement pertinent notamment associé à un langage d'aspect pour la composition dynamique [3] d'une part et un ensemble de composants orientés services pour la découverte dynamique de dispositifs [4] d'autre part. En fonction des contextes, et notamment des dispositifs disponibles, différents aspects peuvent alors être sélectionnés, appliqués et tissés pour adapter l'application. WComp [1] est un environnement de prototypage d'applications sensibles au contexte basé sur ces concepts.

Son architecture s'organise autour de *Containers* et de *Designers*. L'objectif des Containers est de prendre en charge dynamiquement les services systèmes requis par les composants d'un assemblage, comme l'instanciation, la désignation, la destruction de composants logiciels fonctionnels et de liaisons. Les *Designers* permettent la manipulation dynamique des assemblages de composants au travers les containers qui les gèrent. Un *Designer* graphique d'architecture comme Bean4WComp permet par exemple de composer manuellement des assemblages de composants à partir d'une représentation graphique des flots d'événements. Il est particulièrement adapté à la description de l'application. Un *Designer* d'aspects d'assemblage comme ISL4WComp permet quant à lui, par le biais d'une évolution du langage ISL (Interaction Specification Language), de décrire des schémas d'interaction. Ces derniers sont alors sélectionnables, applicables et tissables et permettent ainsi d'adapter dynamiquement l'application précédemment décrite à son contexte.

Enfin, dans un soucis permanent de mise en œuvre et d'expérimentation, WComp est au cœur d'un environnement expérimental original : une plate-forme d'étude des usages des équipements informatiques mobiles en environnement simulé, appelé « Ubiquarium¹ Informatique ». L'Ubiquarium est constitué pour cela de divers dispositifs, comme autant de services découvrables et composables dynamiquement par WComp. Ces dispositifs peuvent être à la fois des dispositifs virtuels (objets d'une scène 3D dans laquelle l'utilisateur est immergé), et des dispositifs réels portés sur lui ou présents dans son environnement. Un tel environnement est un cadre idéal à l'évaluation des nouvelles applications de l'informatique mobile et ambiante telles que les usages des ordinateurs portés ou « wearable computers ».

Références

[1] Daniel Cheung, Jean-Yves Tigli, Stéphane Lavirotte et Michel Riveill. « Wcomp: a Multi-Design Approach for Prototyping Applications using Heterogeneous Resources », *Proceedings of the 17th IEEE International Workshop on Rapid System Prototyping*, Chania, Crete, juin 2006. IEEE Computer Society Press.

¹ du Latin *Ubique*, en toute chose et tout être, avec le suffixe *rium* signifiant lieu ou structure. Donc Ubiquarium Informatique : « lieu ou structure dans laquelle l'informatique est en toute chose et tout être »

[2] Daniel Cheung, Mireille Blay-Fornarino, Jean-Yves Tigli, Stéphane Lavirotte et Michel Riveill. « Adaptation dynamique d'assemblages de dispositifs par des modèles ». Dans les *Actes de la 2^{èmes} Journées sur l'Ingénierie Dirigée par les Modèles (IDM)*, juin 2006.

[3] Daniel Cheung-Foo-Wo, Mireille Blay-Fornarino, Jean-Yves Tigli, Anne-Marie Dery, David Emsellem et Michel Riveill. « Langage d'aspects pour la composition dynamique de composants embarqués ». A paraître dans la *Revue L'Objet*, édition Hermès.

[4] UPnP Forum, "Understanding UPnP : A White Paper", Juin 2000, http://www.upnp.org/download/UPNP_UnderstandingUPNP.doc

3.7 Franck Tarpin-Bernard - AMF pour la plasticité ?

Description

AMF est un modèle multi-agents multi-facettes permettant de structurer le contrôle d'une application. Il couvre les 3 composants centraux du modèle Arch (adaptateur du noyau fonctionnel, contrôleur de dialogue, présentation logique). Ces points forts sont :

- une représentation graphique standardisée mais extensible des mécanismes de contrôle
- une aide à la structuration par la définition de facettes spécialisées
- une utilisation durant tout le processus depuis la conception jusqu'à l'exécution grâce à un moteur
- un métamodèle XML décrit les composants AMF et leurs possibles relations. Un modèle permettant de décrire l'architecture d'une application réelle (c'est le modèle résultant qui est utilisé à l'exécution par le moteur)
- le support de différents niveaux d'adaptation (classes applicatives, facettes, ports, administrateurs...)

En résumé, de part sa position centrale dans l'Arch et ses caractéristiques propres, AMF peut permettre d'agrèger les avantages des différentes approches.

Références

Tarpin-Bernard F., David B.T. AMF : un modèle d'architecture multi-agents multi-facettes. In *Technique et Science Informatiques*, vol. 18, N° 5, 1999.

Masserey G., Tran C. D., Samaan K., Tarpin-Bernard F., David B., Environnement de conception et développement d'applications interactives basées sur l'architecture AMF, In *proceedings of IHM2005*, International Conference Proceedings Series, ACM, 2005, Toulouse, France. September 2005.

Samaan K., Tarpin-Bernard F. Task models an Interaction models in a Multiple User Interfaces generation process. In *Proc. of Tamodia'2004*, pp. 137-144.

Tarpin-Bernard F., David B.T., Primet P. Frameworks and patterns for synchronous groupware: AMF-C approach. In *Proc. of EHCI'98*, pp. 225-241.

3.8 Gaëlle Calvary - les comets : des widgets pour la plasticité

Travail de Olfa Daassi (Thèse 4^{ème} année) et Alexandre Demeure (thèse 3^{ème} année), encadrés par Gaëlle Calvary

Description

Une comète est un widget façonné pour la plasticité des IHM. On entend par plasticité, la capacité d'une IHM à s'adapter à son contexte d'usage (<utilisateur, plate-forme, environnement>) dans le respect de son utilisabilité. Une comète est, par définition, un interacteur méta-décrit, éventuellement polymorphe et auto-adaptable. La méta-description porte sur la tâche utilisateur qu'il supporte, les concepts du domaine qu'il manipule, les espaces de travail qui le composent, les présentations dont il dispose, les ressources qu'il requiert, etc. Le polymorphisme peut agir à tout niveau d'abstraction, couvrant en particulier les tâches polymorphes de Stephanidis. En un mot, c'est l'octopus qui est ici embarqué : une comète est une toile de modèles, chaque modèle et mapping se décrivant. Un inspecteur est présenté permettant, tels des rayons X, d'inspecter l'état et le potentiel de la comète. L'inspection est aujourd'hui limitée au polymorphisme. L'inspecteur offre un ensemble d'opérations élémentaires permettant de substituer une présentation par une autre, d'ajouter une présentation, etc.

Une première implémentation d'une boîte à outils de comets est présentée. Nous appelons Coolkit cette toolkit. Elle démontre le polymorphisme d'une comète ainsi que l'inspecteur de comètes (substitution/suppression/ajout d'une présentation avec conservation de l'état d'interaction).

Référence

Calvary, G., Daassi, O., Coutaz, J., Demeure, A. Des Widgets aux comètes pour la Plasticité des Systèmes Interactifs. Revue d'Interaction Homme-Machine, Europa, Paris, Volume 6, n°1, ISSN 1289-2963, pp 33-53, 2005

Calvary, G., Coutaz, J., Daassi, O., Balme, L., Demeure, A. Towards a new generation of widgets for supporting software plasticity: the « comet », EHCI-DSVIS'2004, The 9th IFIP Working Conference on Engineering for Human-Computer Interaction Jointly with The 11th International Workshop on Design, Specification and Verification of Interactive Systems, Bastide, R., Palanque, P., Roth, J. (Eds), Lecture Notes in Computer Science 3425, Springer, ISSN 0302-9743, Hamburg, Germany, July 11-13, 2004, pp 306-323

Demeure, A., Calvary, G., Coutaz, J., Vanderdonck, J. The Comet Inspector: Manipulating Multiple User Interface Representations Simultaneously, Proc. of 6th Int. Conf. on Computer-Aided Design of User Interfaces CADUI'06 (Bucharest, 6-8 June 2006), Chapter 13, Springer-Verlag, Berlin, 2006, pp. 167-174.

3.9 Anas M. Hariri - Principes de base d'une démarche de génération d'IHM plastiques

Travail de thèse d'Anas Hariri, encadré par Christophe Kolski

Description

On s'intéresse dans le cadre de la thèse à de telles interfaces, qui doivent avoir la capacité de s'adapter de manière dynamique à leur contexte d'interaction en découvrant et utilisant des informations contextuelles, telles que la localisation de l'utilisateur, la date et l'heure, la proximité d'autres utilisateurs, les ressources disponibles, les caractéristiques de la plateforme cible, les conditions ambiantes, etc. Durant l'exécution, il s'agit de répondre dynamiquement aux modifications contextuelles dans le respect de l'utilité et de l'utilisabilité. En partant du concept d'IHM plastique, notre contribution consiste à générer une telle IHM à partir de son modèle abstrait spécifié déjà dans une méthode de spécification et conception de systèmes interactifs, en utilisant la technique des patrons de conception au niveau du passage à l'interface concrète, et en intégrant une architecture de l'interface basée sur les composants métier logiciels qui seront utilisés comme une des solutions de l'adaptation dynamique au contexte d'usage.

Références

1. Hariri, M.A., Tabary, D., Kolski, C. Plastic HCI generation from its abstract model. In *Proc. of IASTED-HCI Int. Conf. on Human-Computer Interaction* (November 14-16, 2005, Phoenix, USA), ACTA Press, Anaheim, USA, 2005, pp. 246-251.
2. Hariri, M.A., Tabary, D. and Kolski, C. Model Derivation Principles for Specification and Design of Multi-Platform HCI. In *Proc. of Int. Conference on Machine Intelligent ACIDCA-ICMI'2005* (November 5-7, 2005, Tozeur, Tunisia).
3. Tabary, D., Abed, M. software environment task object oriented design (ETOOD). *Journal of Systems and Software*, Vol. 60, 2002, pp. 129-140.
4. Kolski, C., Petit-Roze, C., Anli, A., Abed, M., Grislin-Le Strugeon, E., Ezzedine, H., Trabelsi, A. *La plasticité vue sous l'angle de la personnalisation ou selon les besoins vis-à-vis de l'information transport*. Journées thématiques de l'AS Plasticité du RTP 16 IHM, Namur, Belgique, août 2004.

4. Positionnement comparatif des travaux

Une jolie complémentarité s'est dégagée des présentations. Merci à tous pour le travail réalisé. Merci, en particulier, à Franck pour son aide dans leur rédaction.

D'un point de vue méthodologique :

- Articulation des processus de conception IHM et GL :
L'approche par patrons permet de formaliser un parallèle entre des étapes de conception de l'IHM et du GL. L'intérêt de cette articulation est une meilleure coordination entre les différents acteurs de la conception d'un système. Ces parallèles doivent être établis avec les différents modèles et approches impliqués dans la conception et l'implémentation des formes d'interactions.
- Modélisation et conception des techniques d'interaction :
Parmi les travaux présentés, plusieurs constituent des supports à la description des formes d'interaction : ASUR, ASUR-IL, WComp, AMF, ICARE, Comet, mais aussi KMADe.(non présenté, mais évoqué). Les approches présentées diffèrent parce qu'à la base les objectifs ne sont pas les mêmes ! Les aspects traités ne sont donc pas tout à fait les mêmes en couverture et focus. En conséquence, leur positionnement relatif était intéressant.
 - Complémentarité
 - WComp / ASUR : WComp est centré sur les dispositifs et leurs capacités de communication. ASUR est initialement centré sur les aspects de l'interaction ayant un impact sur l'utilisateur. Ces deux approches semblent pouvoir s'enrichir mutuellement.
 - ASUR-IL / AMF : quel lien existe entre les notions de V et C de ASUR-IL et la notion facettes Modèle ? Les types d'administrateurs d'AMF ne constituent ils pas un complément de caractérisation des C et V de ASUR-IL ?
 - ASUR+ASUR-IL / ICARE : les relations ASUR+ASUR-IL sont des composants langages ICARE. Les dispositifs ICARE sont des Adaptateurs ASUR+ASUR-IL. Mais où placer la frontière entre drivers (membre de l'adaptateur et des dispositifs) et API (C de ASUR-IL, composant ICARE langage) ? De plus, les composants combinaisons ne sont pas exprimés dans ASUR+ASUR-IL.
 - Articulation
 - ASUR-IL / AMF : AMF couvre les 3 étapes centrales de l'ARCH, ASUR-IL couvre les 3 étapes de la pile de droite. L'intérêt de et les principes de l'enchaînement de ces deux modèles doivent être analysés.
 - Comet / MAD : une Comet "feuille" couvre fonctionnellement une tâche élémentaire. Les opérateurs entre tâches sont aussi des comets. Comment faire correspondre une hiérarchie de Comets avec un arbre de tâche ? Cet arbre des tâches est-il une vision système « du » modèle des tâches ? Comment embarquer les critères d'ergo (compatibilité / tâche, etc.).
 - Comet / AMF : une comet rend une tâche utilisateur. Elle peut être polymorphe (ie douée de présentations multiples). Il serait intéressant d'étudier dans AMF comment mettre en œuvre ce polymorphisme ainsi que les mécanismes d'adaptation. Plusieurs solutions sont possibles : soit les agents AMF se « branchent » sur des comets (open AMF), soit les comets sont modélisées en agents AMF (close AMF). A noter que les Comets « opérateurs entre tâches » constituent un cas particulier de comets et semblent plus proches des administrateurs, donc à l'intérieur des agents AMF.
 - AMF / MAD : étant donné les articulations possibles entre Comet / MAD et Comet / AMF, le lien AMF / MAD constitue un élément d'articulation à approfondir.
 - Comet / ICARE : une comet rend une tâche utilisateur, via une ou plusieurs présentations. Une présentation est vraisemblablement un composant langage en terme ICARE. Mais les comets « opérateur » ont probablement un lien avec les composants de combinaisons ICARE. De plus, ICARE fait focus sur la multimodalité alors que les comets se focalisent sur la méta-description (dont la qualité – les propriétés CARE en seraient un aspect) et le polymorphisme. Donc complémentarité nette. Autre aspect distinct : ICARE parle de Vue et Input au lieu de Abstraction et Présentation : quelle différence ?
 - Comet / ASUR+ASUR-IL : les ressources d'interaction des comets pourraient être décrites avec ASUR+ASUR-IL lorsqu'il s'agit d'une forme d'interaction mixte.
 - ASUR / MAD : un diagramme ASUR pourrait être un ensemble de tâches et sous-tâches MAD à réaliser en parallèle : il n'y aurait pas plusieurs sous-tâches impliquant l'utilisateur à réaliser en séquence ou de manière alternative. Il faudrait alors

caractériser le parallélisme en termes de CARE, ce qui favoriserait le lien avec les composants ICARE.

D'un point de vue de l'architecture logicielle conceptuelle :

Pour faciliter le développement de formes d'interactions, une conception de l'architecture logicielle adaptée est requise. Parmi les modèles présentés, certains contribuent à cet aspect et présentent des avancées à exploiter :

- On verrait bien AMF au centre pour la force de son contrôleur et l'ouverture de ses facettes.
- Les comets pourraient être modélisées en AMF. Leur capacité d'autodescription (notamment en terme d'utilisabilité) et leurs liens avec les modèles (tâches, contexte...) pourraient faire l'objet de facettes spécifiques (User, Platform, Adaptation, etc.)
- La multimodalité d'ICARE pourrait être prise en compte par de nouveaux administrateurs de contrôle (redondance, équivalence...) et probablement des agents spécifiques.
- Les comets elles-mêmes seraient assemblées selon des schémas ICARE et/ou les composants ICARE seraient encapsulés en des comets (cf la méta-description de la comet).
- L'historique d'Anas pourrait être pris en charge par une facette AMF dédiée.

D'un point de vue implémentatif :

Certains outils méthodologiques présentés sont associés à une solution logicielle permettant le déploiement des modèles développés. De la même manière que pour les approches méthodologiques, des points d'ancrages semblent intéressants à étudier au niveau des supports au développement.

- La robustesse et les avancées faites autour de WComp font de cette plateforme la plus mure des plateformes développées dans le groupe. On verrait donc bien partout des composants Wcomp pour leur ouverture et la prise en compte du monde physique.
- Le moteur AMF peut sans doute faire l'objet d'une implémentation WComp, ce qui lui conférerait la puissance sous jacente, spécialement en terme d'ouverture
- Les comets pourraient être implémentées en AMF, bénéficiant alors de la mise en œuvre Wcomp
- AMF / ISL : Le langage ISL pourrait être embarqué dans les administrateurs AMF pour compléter les opérateurs déjà envisagés.

Du point de vue de l'évaluation

- Analyse de l'utilisabilité : Pour le moment les outils présentés offrent assez peu de réponse. ASUR peut servir de support à l'analyse de propriétés ergonomiques spécifiques aux systèmes mixtes. Les comets ont également pour vocation d'offrir un tel support en embarquant dans leur description la *valeur* qu'elles garantissent.
- Analyse de propriétés logicielles : le constat est similaire. Il serait toutefois intéressant d'approfondir cet aspect afin, par exemple, de pouvoir certifier que des techniques d'interaction garantissent leur adaptabilité à tel ou tel contexte.

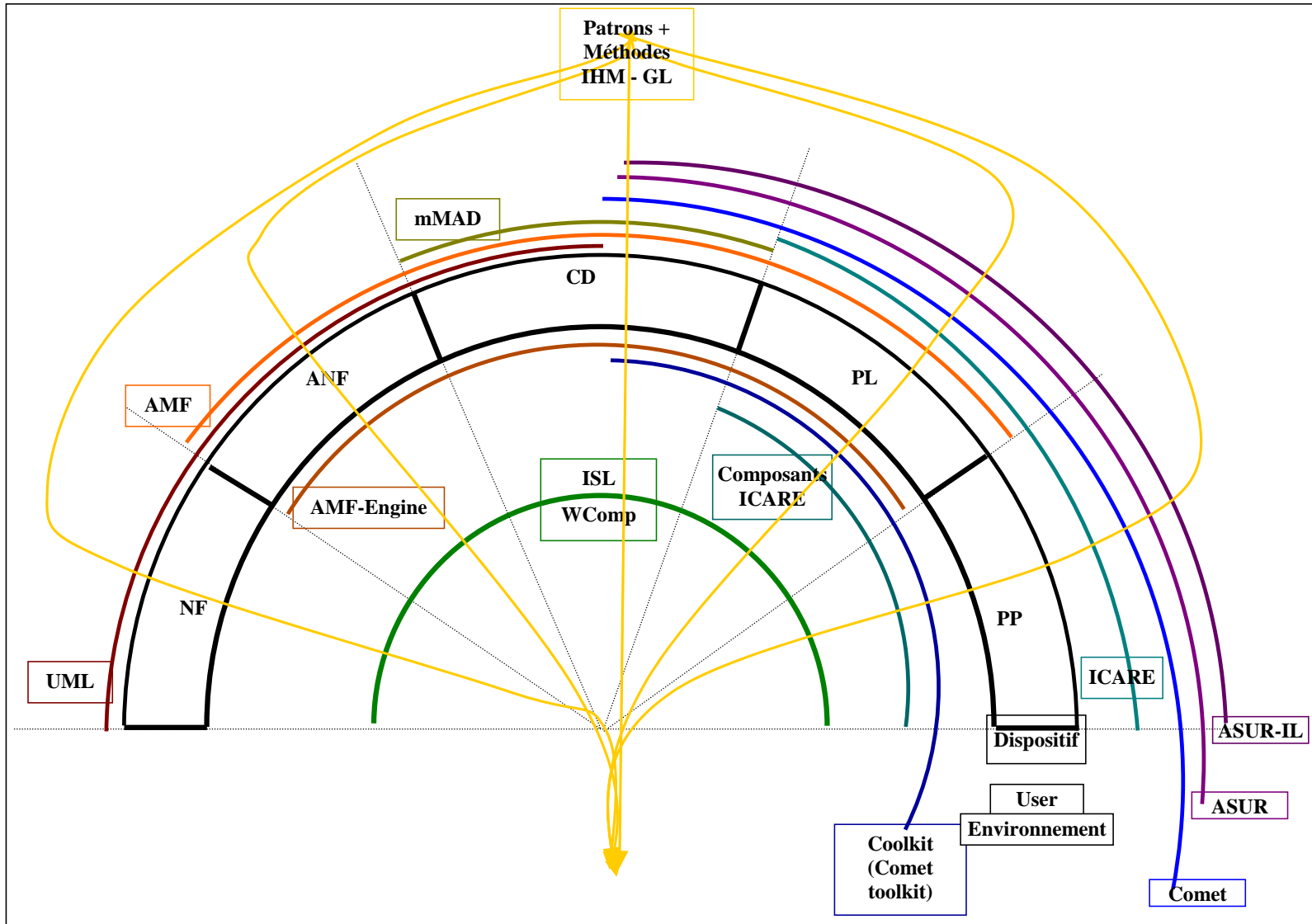
Le tableau ci-après dresse une synthèse des modèles et approches présentés au sein du groupe. Pour chaque approche sont indiqués :

- la phase du cycle de développement visée ;
- les apports ;
- l'existence ou non d'un méta-modèle ;
- l'existence ou non d'un éditeur pour la manipulation du modèle ou de la plateforme ;
- les transitions existantes ou en cours avec d'autres modèles ;
- la possibilité d'exprimer des aspects dynamiques ;
- l'existence d'un support à l'étude de propriétés ergonomiques ;
- des commentaires additionnels.

En complément une représentation graphique de ces différentes approches autour du modèle ARCH a été proposée. L'intérêt est de mettre en évidence les points d'ancrage et/ ou les redondances entre les différentes contributions des membres du groupe. Au milieu du schéma sont rappelés les éléments du modèle ARCH. En-dessous figurent les approches constituant des supports au déploiement / développement de formes d'interactions mixtes, adaptables et en évolution. Au-dessus figurent les approches de modélisation / conception développées par les membres du groupe.

Modèle	Phase	Point(s) fort(s)	Méta-Modèle	Editeur	Transformations	Dynamisme	Propriétés	Commentaires
ASUR	Analyse/Conception	Description de l'interaction utilisateur/système avec objets physiques et numériques pour une tâche et dans un contexte défini et figé	oui	oui	vers ASUR-IL de KMADe ?		oui	travail en cours entre MAD (K-MADe) et ASUR
ASUR-IL	Conception	Présenter composants et assemblage de composants pour un diagramme ASUR pour un couplage entre conception et implémentation	semi		de ASUR vers Wcomp vers ICARE ?			liens avec MVC
ICARE	Conception / Protot	Prise en compte des propriétés CARE dans un contexte figé	oui	semi	de ASUR			liens possibles avec AMF de schémas CARE vivants au runtime
AMF	Conception / Runtime	Modélisation du contrôle de l'interaction Patterns (sans schéma AMF) ≠ niveaux d'adaptation	oui	oui	vers COMET ? vers Wcomp ?	oui		
Comet	Conception / Exécution	Auto-description Éventuellement polymorphisme IHM multifacettes et inspection à la volée des différentes facettes	oui	Oui (Toolkit)	oui	oui	oui	Auto-description fonctionnelle pour l'instant, à enrichir des modèles de contexte et d'utilisabilité
ISL	Conception / runtime dans NOAH	expression de contrôleur de dialogue, support à l'expression/analyse de flots de données règle de fusion de schéma ISL, dynamique de la mise à disposition des schémas,	oui	NOAH .NET Wcomp		oui	oui mais pas IHM	

Modèle	Phase	Point(s) fort(s)	Méta-Modèle	Editeur	Transformations	Dynamisme	Propriétés	Commentaires
				?				
Bean4W Comp	Conception / runtime (plateforme)	Modele vivant / simplicité / runtime / Support efficace au prototypage rapide découverte automatique de ressources d'interaction (Physique ou numérique)	oui	Wcomp		oui		évolution du modèle à composants (ex : connecteurs complexes)
Amusing	conception / vivant à l'exec.	Assemblage d'interface abstraite partielle + liaison avec composants métiers NF		Composants IHM abstrait	CARE ?			
mMAD	Analyse/Conception	Description d'activité avec concepts manipulés Simulation	oui	oui		semi		exécutable mais pas connecté à l'IHM (simulation dynamique d'un scénario sur le modèle)
UML	Analyse/Conception	Standard de modélisation en Génie Logiciel	oui	oui	avec ASUR avec CTT (ou K-MADE)			
LAMIH	Conception /	Partit d'un modèle abstrait d'interface déjà spécifié () Génération automatique d'interfaces Modèle vivant à l'exécution Mise à jour des composants à l'exécution (apprentissage)	oui	non		oui	implicite	apprentissage : apprendre par rapport aux réactions de l'utilisateur, reconfiguration



5. Avenir du groupe

La réunion a très bien fonctionné. Il faut aller plus loin en conservant si possible le « noyau » dur formé à Sophia . Les leviers possibles sont :

- de nouvelles réunions : nous proposons les 12, 13 et 14 mars 2007 à Grenoble
- le co-encadrement de stagiaires
- des publications communes dans les communautés IHM et IDM. IHM : la revue RIHM ? IDM : Mireille ?

Le groupe CESAME est inscrit à l'action IDM-IHM.