

# LES PROPRIETES “CARE” DANS LES INTERFACES MULTIMODALES

Joëlle Coutaz, Laurence Nigay

Laboratoire de Génie Informatique-Institut IMAG

BP 53, 38041 Grenoble Cedex 9

Tel. 76 51 48 54, 76 51 44 40

E-mail : Joelle.Coutaz@imag.fr, Laurence.Nigay@imag.fr

**Résumé :** En réponse à l’avancée des nouvelles technologies d’interaction, il convient d’organiser le savoir dans des référentiels en vue de classer, comparer, concevoir et évaluer les interfaces existantes ou à venir. Nous présentons ici une brève revue des espaces actuels. Une analyse critique nous conduit ensuite à proposer une nouvelle contribution : l’espace pipe-line qui couvre à la fois le point de vue de l’utilisateur et celui du concepteur de logiciel. Les notions de langage d’interaction et de dispositifs physiques servent de point de contact entre ces deux perspectives. Nous montrons que les langages d’interaction et les dispositifs physiques entretiennent des relations qui permettent de caractériser les nouvelles interfaces. Il s’agit des propriétés CARE : Complémentarité, Assignation, Redondance et Equivalence. Ces notions ont été présentées dans [MARTIN 94]. Nous leur donnons ici une facture formelle et montrons en conclusion leur utilité pour classer ou évaluer des systèmes interactifs.

**Mots clés :** Complémentarité, assignation, redondance, équivalence, espace de conception, interface multimodale, dispositif physique d’entrée/sortie, langage d’interaction, classification.

## 1. Introduction

La communication homme-machine est un domaine évolutif tiraillé entre la pratique créative dirigée par la technique et l’approche conceptuelle et mesurée de la théorie. Pratique et théorie sont les deux approches complémentaires qui servent de piliers aux démarches scientifiques. Nous constatons aujourd’hui une explosion de techniques de communication ouvrant des voies nouvelles dont on ne cerne pas les limites. Prenons pour preuve le flou d’une terminologie mal assise qui fléchit au gré de la mode : interfaces multimédia et multimodales, réalités virtuelles, artificielles et augmentées.

Il convient dans ces conditions d’organiser le savoir expérimental dans des cadres aussi rigoureux que possible en vue de classer, comparer, concevoir et évaluer les interfaces existantes ou à venir. Au paragraphe 2, nous présentons une brève revue des espaces actuels. Une analyse critique nous conduit au paragraphe 3 à proposer une nouvelle contribution : l’espace pipe-line qui couvre à la fois le point de vue de l’utilisateur et celui du concepteur de logiciel. Les notions de langage d’interaction et de dispositifs physiques servent de point de contact entre ces deux perspectives. En 4, nous montrons que les langages d’interaction et les dispositifs physiques entretiennent des relations qui permettent de caractériser les nouvelles interfaces. Il s’agit des propriétés CARE : Complémentarité, Assignation, Redondance et Equivalence. Ces notions ont été présentées dans [MARTIN 94]. Nous leur donnons ici une facture formelle et montrons en conclusion leur utilité pour classer ou évaluer des systèmes interactifs. On trouvera dans [NIGAY 94] l’analyse de leur incidence sur les architectures logicielles et les solutions apportées.

## 2. Revue et analyse critique des référentiels actuels

La plupart des espaces de conception dissocient l'interface d'entrée de l'interface de sortie. D'autres ne considèrent qu'une seule des deux dimensions et se distinguent par le niveau d'abstraction considéré. En particulier, certains ne couvrent que les dispositifs physiques (bas niveau d'abstraction) alors que d'autres intègrent les traitements de haut niveau.

Les premières tentatives de taxonomie concernent les dispositifs d'entrée. Dans cette lignée, il convient de citer la grille d'analyse de Buxton [BUXTON 83] et la théorie des dispositifs d'entrée de Card, Mackinlay et Robertson [CARD 90]. Selon Buxton, un dispositif d'entrée est un transducteur de propriétés physiques. La richesse du dispositif est fonction du nombre de propriétés physiques captées et pour chaque propriété, le nombre de dimensions spatiales. Le type (direct ou indirect) de la capture traduit le degré d'engagement physique de l'utilisateur.

La taxonomie de Buxton ne concerne que les dispositifs continus actionnables par le geste. La variabilité des niveaux d'abstraction n'est pas davantage considérée. Par exemple, en vision par ordinateur, un niveau d'abstraction pertinent est la détection des contours et leur mouvement comme dans le "videoplace" de Krueger [KRUEGER 91], ou mieux encore, la position (x, y, z) de chaque doigt comme il faudrait le faire pour le "bureau digital" d'EuroPARC [WELLNER 93]. J. Crowley traduit l'existence de niveaux d'abstraction avec la notion de "capteur logique" [CROWLEY 94]. Une argumentation similaire vaut pour le microphone. La théorie des dispositifs d'entrée de Card, Mackinlay et Robertson traduit la notion de niveau d'abstraction.

Mackinlay et ses coauteurs systématisent la description des dispositifs d'entrée au moyen d'une algèbre simplifiée. L'avantage est double : élargir le champ de conception à la prospection de nouveaux dispositifs et ouvrir la voie à l'évaluation du pouvoir d'expression des dispositifs en relation avec la tâche. La démonstration présentée par Mackinlay, Card et Robertson, dans leur article [CARD 90] ne concerne que les dispositifs actionnables explicitement par l'utilisateur. Cette théorie qui rend explicite la notion de domaines de valeurs, contribue cependant à attirer l'attention du concepteur. Par exemple, dans le cas du "bureau digital", il conviendrait que le système de vision par ordinateur, partant de l'ensemble des données d'entrée de type pixels, soit capable en sortie de modéliser les doigts et la main en accord avec le type de tâches envisagées. La taxonomie de Foley, plus ancienne, part de cette notion de tâche [FOLEY 84].

A l'inverse des interfaces d'entrée, on a porté peu d'efforts sur la classification des interfaces de sortie. La taxonomie sur les "modalités pures" d'Ole Berntsen est une exception. En accord avec la métaphore des distances sémantiques et articulatoires de la théorie de Norman [NORMAN 86], Berntsen s'interroge sur le problème de la représentation optimale des informations que le système et l'utilisateur sont amenés à échanger au cours de leur tâche commune. Dans cet espace problème, Berntsen centre son étude sur la représentation des expressions de sortie qu'il appelle modalités représentationnelles [BERNTSEN 93]. Un graphe, un texte, une icône sont de telles modalités. Ici, modalité doit se comprendre comme signifiant. Il ne s'agit pas des capacités sensorielles humaines telles qu'on l'entend en psychologie. L'objectif est de définir une théorie qui permette au concepteur d'identifier les modalités (les signifiants) qui conviennent à la représentation des concepts du système (les signifiés).

Tout comme la taxonomie de Mackinlay, la classification de Berntsen se veut "généralisatrice" : elle s'appuie sur une classe de modalités élémentaires dites "modalités pures" combinables en représentations plus complexes. Chez Berntsen, une modalité se définit par le doublet <média, profil>. Le média désigne le support d'expression en

relation avec les capacités sensorielles humaines. Il prend sa valeur dans l'ensemble {graphique, son, toucher}. Le profil regroupe un ensemble de propriétés : statique ou dynamique, linguistique ou non, analogique ou non (qui entretient ou non un rapport de ressemblance avec la réalité), arbitraire ou non (qui fonctionne ou non en dehors d'un système conventionnel).

Pour résumer, la résolution du problème de la correspondance entre signifiants et signifiés repose sur l'existence d'une taxonomie solide qui offre des possibilités d'analyse. La théorie de Mackinlay, Card et Robertson vise cet objectif pour les dispositifs d'entrée. Bernsen poursuit un but similaire pour les sorties. Si la théorie des premiers est dirigée par la technologie, la taxonomie de Bernsen est davantage liée aux capacités cognitives humaines.

Nous venons de présenter des taxonomies qui concernent soit les entrées soit les sorties. L'espace de Frohlich [FROHLICH 91] et MSM [COUTAZ 93] recouvrent les deux directions mais l'un et l'autre sont bornés par des points de vue restrictifs : le premier, centré sur l'utilisateur, ne permet pas de relier les décisions de conception à leurs conséquences sur la réalisation informatique. Inversement, MSM, qui opte pour une perspective système, masque le point de vue utilisateur. Le modèle pipe-line présenté ci-dessous vise à concilier les points de vue utilisateur et système.

### **3. Un référentiel intégrateur : l'espace pipe-line**

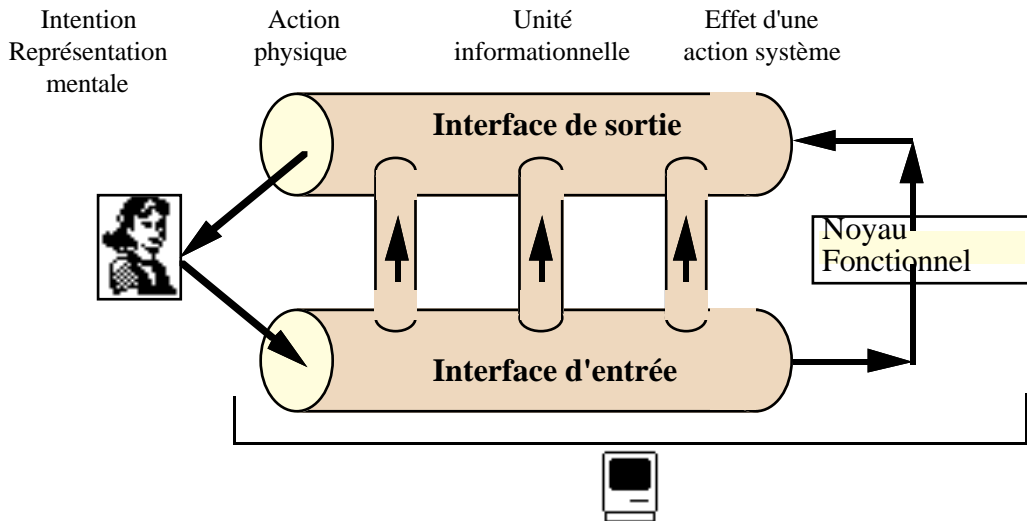
Le modèle pipe-line repose, comme la théorie de Norman et le Modèle du Processeur Humain, sur le principe des flux d'information entre les acteurs communicants. Alors que Norman détaille les activités mentales de l'utilisateur sous forme d'étapes et esquisse les traitements du système, le modèle pipe-line offre une analyse complémentaire : les activités mentales y sont esquissées et les traitements informatiques y sont détaillés et organisés en étapes.

Comme le montre la figure 1, l'utilisateur traduit une intention en actions physiques. Une fois exécutée, une action physique ou un ensemble d'actions physiques peut être acquis puis interprété par le système pour former une unité informationnelle. A partir d'une unité informationnelle, le système déduit une action système dont l'exécution, par le système, provoque un effet. Partant d'un effet, le système choisit la ou les unités informationnelles aptes à représenter l'effet en fonction de la situation. Cet ensemble d'unités informationnelles est traduit en actions physiques réalisées cette fois-ci par le système. L'utilisateur est alors en mesure de percevoir puis d'interpréter les actions physiques pour élaborer une représentation mentale de l'état et/ou du fonctionnement du système.

Pour simplifier l'exposé, nous avons considéré que le flot des informations suivait un cheminement séquentiel et symétrique allant des intentions aux effets, puis des effets aux représentations mentales. Comme le montre la figure 1, il existe d'autres cheminements, sortes de courts-circuits entre les tuyaux d'entrée et de sortie qui traduisent des retours d'information à différents niveaux d'abstraction : niveaux lexical, syntaxique et sémantique. (On retrouve ces courts-circuits dans le modèle de Rasmussen sur les comportements cognitifs.) Deux phénomènes viennent se greffer sur ces cheminements : le parallélisme et la fusion/fission. Dans [NIGAY 94], nous montrons comment le modèle pipe-line affine ces notions issues de MSM.

Ayant esquissé le principe de fonctionnement de Pipe-line, il convient maintenant d'en préciser les termes : une *action physique* est une opération qui modifie l'état d'un dispositif physique. Un *dispositif physique* est, comme la souris, un transducteur de propriétés physiques. Notre notion de dispositif d'entrée est identique à celle de Mackinlay ; celle de dispositif de sortie couvre la notion de média chez Bernsen. Une

*unité informationnelle* regroupe un ensemble indivisible d'informations qui, dans le système, modélise un concept du domaine de la tâche. Parce qu'elle fait sens, une unité informationnelle est qualifiée de "représentation au niveau d'abstraction le plus haut". Une *action système* est une unité de traitement atomique : elle est indivisible. Un *effet* est la différence entre l'état interne initial (c.-à-d. l'état interne avant l'exécution de l'action système) et l'état interne final (c.-à-d. l'état interne après l'exécution de l'action système).



**Figure 1:** Le principe du modèle Pipe-line.

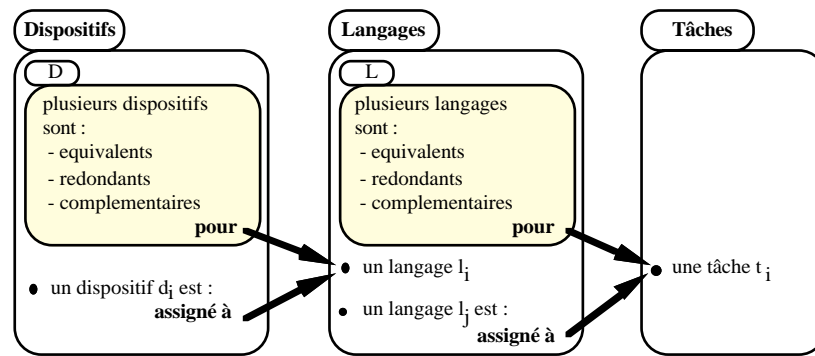
On appelle *acquisition*, la fonction qui fait passer des actions physiques utilisateur aux unités informationnelles d'entrée. *Restitution* est la fonction qui, des unités informationnelles de sortie, produit des actions physiques perceptibles pour l'utilisateur. L'une et l'autre dirigent leurs calculs en fonction du dispositif physique et du langage d'interaction courants. Nous appelons *langage d'interaction* un système conventionnel structuré de signes qui assure une fonction de communication entre un système informatique et un utilisateur. De manière plus formelle, un langage d'interaction se définit par un vocabulaire d'éléments terminaux et une grammaire.

Le langage d'interaction et le dispositif physique forment les deux facettes complémentaires d'une expression : le langage définit la structure c'est-à-dire les relations de dépendance qu'entretiennent les constituants de l'expression ; le dispositif définit la réalité physique observable. Si l'on adopte la terminologie de Hejmslev [HEJMSLEV47], le langage d'interaction définit la forme d'une expression tandis que le dispositif physique en définit la substance. De fait, les notions de langage d'interaction et de dispositif définissent deux points de contact entre le système et l'utilisateur. Au paragraphe suivant nous étudions les relations qu'ils entretiennent via les propriétés CARE.

#### 4. Les propriétés CARE

La figure 2 montre les relations possibles entre dispositifs, langages d'interaction et tâches. Ces relations peuvent être permanentes ou temporaires, totales ou partielles. Une relation est permanente si elle est vérifiée quel que soit l'état du système ; elle est temporaire dans le cas contraire. Une relation entre un dispositif et un langage est totale si elle est vérifiée pour toutes les expressions de ce langage (sinon, elle est dite partielle). Une relation entre un langage et les tâches que le système permet d'accomplir est totale si elle tient pour toutes les tâches (sinon, elle est partielle). Dans cette discussion, les tâches considérées sont élémentaires au sens de [BALBO 93] : celles-ci ne peuvent être décomposées que sous forme d'actions physiques.

Il convient maintenant de définir plus formellement les propriétés CARE. Pour cela nous notons  $s$  un état du système,  $D$  l'ensemble des dispositifs physiques de même direction (entrée ou sortie) offerts par le système,  $D$  un sous-ensemble non vide de  $D$ ,  $d$  un élément de  $D$  ou  $D$ ,  $L$  l'ensemble des langages d'interaction de même direction offerts par le système,  $L$  un sous-ensemble non vide de  $L$ ,  $l$  un élément de  $L$  ou de  $L$ ,  $T$  l'ensemble des tâches élémentaires réalisables avec le système,  $T$  un sous-ensemble non vide de  $T$ , et  $t$  un élément de  $T$ . Nous commençons par les relations entre langages et tâches puis entre dispositifs et langages.



**Figure 2 :** Les relations CARE (Complémentarité, Assignment, Redondance et Equivalence) entre dispositifs, langages d'interaction et tâches.

*Equivalence de langages pour une tâche  $t$  : L-Equivalence ( $L, s, t$ )*

Les langages d'interaction de  $L$  sont équivalents dans l'état  $s$  du système pour la tâche  $t$ , si  $t$  peut être exprimée au moyen de l'un quelconque des langages de  $L$ . *L-Equivalence* ( $L, s, t$ ) est permanente si elle vraie quel que soit  $s$ . Elle est totale si elle est vraie pour toutes les tâches de  $T$ . Dans MATIS [NIGAY 93], l'utilisateur peut spécifier ses demandes de renseignement sur les horaires d'avion en utilisant la manipulation directe ou le langage naturel. Cette relation est permanente (la demande de renseignement peut toujours se faire via le langage naturel ou la manipulation directe) mais elle est partielle : elle ne tient que pour la tâche de demande de renseignements.

*Assignment de langage à une tâche : L-Assignment ( $l, s, t$ )*

Un langage d'interaction  $l$  est assigné à la tâche  $t$  dans l'état  $s$ , s'il n'existe pas de langages  $l' \neq l$  tel que *L-Equivalence* ( $l', s, t$ ). En d'autres termes,  $t$  ne peut être exprimée qu'au moyen de  $l$ . Dans MATIS, il y a *L-Assignment* permanente du langage de manipulation directe pour les tâches de gestion des fenêtres (changement de taille, fermeture, etc.). Cette assignation n'est que partielle puisque pour la demande de renseignement, il est possible d'employer aussi le langage naturel.

*Complémentarité de langages pour l'expression d'une tâche : L-Complémentarité ( $L, s, t$ )*

Les  $n$  langages  $l_1, l_2, l_n$  de  $L$  sont complémentaires dans l'état  $s$  pour l'expression de la tâche  $t$  si l'expression de  $t$  peut être partitionnée en  $n$  sous-expressions  $exp_1, exp_2, exp_n$ , telles que chaque sous-expression  $exp_i$  est construite dans un et un seul langage  $l_i$  de  $L$  : *L-assignment* ( $l_i, s, exp_i$ ). Les coréférences entre langages illustrent cette situation. Dans MATIS, le langage naturel et la manipulation directe sont partiellement complémentaires mais de manière permanente : cette relation tient quel que soit l'état mais seulement pour les tâches de spécification des champs d'une requête. Articuler la phrase *flights from this city* ou *flights from* accompagné de la sélection de la ville au moyen de la souris est un exemple de complémentarité entre expressions de langages distincts. Les deux expressions sont nécessaires à la constitution de la tâche de spécification "lieu de départ". Notons que la complémentarité autorise un usage synergique ou alterné de langages : il y a toujours combinaison mais les expressions peuvent être produites en parallèle ou de manière séquentielle.

*Redondance de langages dans l'expression d'une tâche : L-Redondance (L, s, t)*

Dans un état  $s$ , il y a redondance entre les langages de  $L$ , si ces langages sont  $L$ -équivalents sur  $s$  et  $t$  et s'ils peuvent être utilisés simultanément ou de manière séquentielle mais contrainte par une fenêtre temporelle. Dans MATIS, la redondance est permanente entre le langage naturel et le langage de manipulation directe mais elle est partielle : elle ne tient que pour les tâches de demande de renseignement. L'énoncé *flights to Boston* et la saisie concurrente et équivalente de *Boston* dans le champ de destination d'un formulaire donne naissance à un seul exemplaire de requête contenant Boston.

Nous traitons maintenant les relations entre les dispositifs et les langages.

*Equivalence de dispositifs pour une expression  $el$  du langage  $l$  : D-Equivalence (D, s, el, l)*

Les dispositifs physiques de  $D$  sont équivalents dans l'état  $s$  pour construire l'expression  $el$  de  $l$ , si  $el$  peut être élaborée au moyen de l'un quelconque des dispositifs de  $D$ . *D-Equivalence (D, s, el, l)* est permanente si elle vraie quel que soit  $s$ . Elle est totale si elle est vraie pour toutes les expressions de  $l$ . Dans MATIS, l'utilisateur peut construire des expressions en langage naturel en utilisant soit le clavier soit le microphone. Cette *D-equivalence* est permanente mais partielle : il n'est pas possible de corriger les erreurs d'énonciation via le microphone (par exemple l'énoncé oral "flights from ... no! I mean : to" n'est pas autorisé alors que la correction d'erreurs de frappe est permise).

*Assignment de dispositif à une expression  $el$  du langage  $l$  : D-Assignment (d, s, el, l)*

Un dispositif  $d$  est assigné à l'expression  $el$  de  $l$  dans l'état  $s$ , s'il n'existe pas de dispositifs  $d' \neq d$  tel que *D-Equivalence (d', s, el, l)*. En d'autres termes,  $el$  ne peut être construite qu'au moyen de  $d$ . L'assignation est totale si elle est vraie pour toutes les expressions de  $l$ . Dans MATIS, il y a *D-Assignment* permanente et totale de la souris au langage de manipulation directe.

*Complémentarité de dispositifs pour la construction d'une expression  $el$  du langage  $l$  : D-Complémentarité (D, s, el, l)*

Les  $n$  dispositifs  $d_1, d_2, \dots, d_n$  de  $D$  sont complémentaires dans l'état  $s$  pour la construction de l'expression  $el$  de  $l$  si  $el$  peut être partitionnée en  $n$  sous-expressions  $expl_1, expl_2, \dots, expl_n$ , telles que chaque sous-expression  $expl_i$  est construite au moyen d'un et un seul dispositif  $d_i$  de  $D$  : *D-assignation (d\_i, s, expl\_i, l)*. La complémentarité est totale si elle tient pour toutes les expressions de  $l$ . *Munix [POIRIER-LEFEBVRE 93]* offre un exemple de complémentarité partielle mais permanente entre le clavier et le microphone mais aussi entre la souris et le microphone pour les expressions du langage naturel dans lesquelles apparaissent des noms d'objet (par exemple des noms de fichiers). (On observe ici l'équivalence du clavier et de la souris pour la tâche de désignation des objets. Cette équivalence est totale mais partielle car elle ne tient pas pour toutes les tâches du système.)

*Redondance de dispositifs dans la construction d'une expression  $el$  du langage  $l$  : D-Redondance (D, s, el, L)*

Dans un état  $s$ , il y a redondance entre les dispositifs de  $D$ , si ces dispositifs sont  $D$ -équivalents sur  $s$  et  $el$ , et s'ils peuvent être utilisés simultanément. Un tel fonctionnement s'applique par exemple en vision stéréoscopique. Un autre exemple possible est le montage "redondant" d'un microphone et d'une caméra qui observe le mouvement des lèvres pour augmenter la robustesse d'un système de reconnaissance de la parole.

## 5. Utilité des propriétés CARE

A l'évidence, les propriétés CARE permettent de comparer des interfaces en termes de langages d'interaction, de dispositifs et de leurs relations. Dans [NIGAY 94], nous montrons plusieurs méthodes de classification selon que l'on considère l'usage

(synergique, concurrent, alterné, exclusif.) et le caractère optionnel ou obligatoire des langages d'interaction et des dispositifs. Les notions de synergie, concurrence, etc. avaient été introduites dans le cadre des journées IHM'93. Nos notions de langage d'interaction et de dispositif permettent d'en préciser la portée.

Le deuxième intérêt des propriétés CARE est leur application à l'évaluation ou à la conception des nouvelles interfaces. Dans [David 93], on montre comment relier la sélection des modalités en fonction du modèle de dialogue. Dans [ABOWD 92], nous structurons l'utilisabilité en robustesse et souplesse. Nous mesurons la souplesse d'un système par sa capacité à offrir ou à faire des choix : absence de préemption, représentation multiple d'un même concept, égale opportunité, adaptativité, multimodalité. Tous ces critères concernent plusieurs niveaux d'abstraction depuis les dispositifs physiques jusqu'au noyau fonctionnel. Or les notions de dispositif physique, de langage d'interaction et de tâche dénotent trois niveaux d'abstraction distincts qui nous permettent d'unifier certains des critères ci-dessus sous la forme de nouvelles relations qualifiantes entre langages et entre dispositifs : les propriétés CARE. Voyons comment nous pouvons a priori les corrélérer aux critères de robustesse et de souplesse.

L'équivalence et l'assignation caractérisent la portée des choix en matière de langage et de dispositif. La première joue en faveur de la souplesse ou de la robustesse (par exemple, dans un milieu bruité, le clavier peut être substitué au microphone), la seconde augmente le caractère préemptif du système (l'utilisateur n'a pas de choix). La redondance et la complémentarité qualifient la combinaison des choix. La première intervient, selon l'intention communicationnelle, comme un facteur de souplesse ou de robustesse. La seconde, qui impose un usage complémentaire de langages et/ou de dispositifs, peut se voir comme un facteur contraignant. Sous cet angle, le paradigme "mets ça là" souvent cité comme modèle d'interaction "naturelle", apparaît d'une utilisabilité douteuse : il requiert la complémentarité de deux langages mais aussi l'assignation de deux dispositifs. En vérité, des expérimentations par Magicien d'Oz montrent que les sujets en phase exploratoire utilisent volontiers l'équivalence et la redondance mais une fois les capacités du système cernées, une large place est faite à l'assignation [MIGNOT 93].

Les caractères partiel et temporaire d'une relation CARE peuvent aussi servir de guide prédictif : une relation qui n'est pas totale ou qui n'est pas permanente indique que l'utilisateur devra se souvenir des contextes particuliers dans lequel elle s'applique.

Les propriétés CARE fournissent un cadre cohérent et suffisant si l'on s'en tient à l'aspect fonctionnel (tâche) du système. Il conviendrait de considérer des propriétés plus fines au niveau des dispositifs physiques comme au niveau des langages afin de mieux cerner les coûts système mais aussi les coûts cognitifs, conatifs et affectifs [DOWELL 89]. C'est ce que nous proposons de discuter lors des journées IHM'94.

## **6. Remerciements**

Ce travail a été partiellement financé par le projet ESPRIT BRA AMODEUS 7040 et le PRC CHM, pôle IM multimodales.

## **7. Références**

[ABOWD 92]

G. Abowd, J. Coutaz et L. Nigay, Structuring the Space of Interactive System Properties, Proceedings of the IFIP TC2/WG2.7 on Engineering for Human-Computer Interaction, Ellivuori, Finlande, (10-14 Août 1992), 113-128.

[BALBO 93]

S. Balbo, J. Coutaz, "Automatic Evaluation in Human Computer Interaction", The Ergonomics Society 1993 Annual Conference, Edinburgh, 13-16 April, 1993.

- [BERNSEN 93]  
N.O. Bernsen, "Structuring the design space", in the InterCHI'93 Adjunct Proceedings, pp. 211-212, Amsterdam, 1993.
- [BUXTON 83]  
W. Buxton, "Lexical and Pragmatic considerations of Input Structure", Computer Graphics, 17(1), 1983, pp. 31-37.
- [CARD 90]  
S. Card, J. Mackinlay, G. Robertson, "The Design Space of Input Devices", in Proc. CHI'90, Seattle, pp. 117-124.
- [COUTAZ 93]  
J. Coutaz, L. Nigay, D. Salber, The MSM framework: A Design Space for Multi-Sensori-Motor Systems. In Human Computer Interaction, 3rd International Conference EWHCI'93, East/West Human Computer Interaction, Moscow. L. Bass, J. Gornostaaev, C. Unger Eds. Springer Verlag Publ., Lecture notes in Computer Science, Vol. 753, 1993, pp.231-241.
- [CROWLEY 94]  
J.L. Crowley, H. Christensen. "Vision as Process: Integration and Control of Real Time Active Vision System" dans Experimental Environments for Computer Vision and Image Processing, Series in Machine Perception and Artificial Intelligence, Vol. 11, World Scientific Pub., H.I Christensen, J.L. Crowley Eds., pp 127-155, 1994.
- [DAVID 93]  
B. David, S. Sadou, P. Spirito, M. Djebali, "Sélection des modalités pour une interface multimodale", Actes du colloque IHM'93, pp. 37-44.
- [DOWELL 89]  
J. Dowell & J. Long, "Toward a conception for an engineering discipline of human factors", Ergonomics, vol. 32, No 11, pp. 1513-1535, November 1989.
- [FOLEY 84]  
J.D. Foley, A. Van Dam, The Fundamentals of Computer Graphics, Addison Wesley, 1984.
- [FROHLICH 91]  
D. Frohlich, "The design space of interfaces, Multimedia Systems, Interaction and Application", Proc. 1rst Eurographics Workshop, Stockholm, Springer Verlag, 1991, pp. 53-69.
- [HEJMSLEV 47]  
L. Hejmslev, Structural Analysis of language, Studia Phonetica, vol. 1, (1947), 69-78.
- [KRUEGER 91]  
M. W. Krueger. "Artificial Reality II". Addison Wesley, 1991.
- [MARTIN 94]  
J.-C Martin, Cadre d'étude de la multimodalité fondé sur les types et buts de coopération entre modalités, Actes de MONTPELLIER'94, l'interface des mondes réels & virtuels, Montpellier, (Février 1994), 97-106.
- [MIGNOT 93]  
C. Mignot, C. Valot, N. Carbonell, An Experimental Study of Future 'Natural' Multimodal Human-Computer Interaction, Adjunct Proceedings of InterCHI'93, Amsterdam, (Avril 1993), 67-68.
- [NIGAY 93]  
L. Nigay, J. Coutaz, A design space for multimodal interfaces: concurrent processing and data fusion, Proceedings of InterCHI'93, Amsterdam, (Avril 1993), 172-178.
- [NIGAY 94]  
L. Nigay, Conception et modélisation logicielles des systèmes interactifs, thèse de l'Université Joseph Fourier, Grenoble, (Janvier 1994), 350 pages.
- [NORMAN 86]  
D. A. Norman, Cognitive Engineering, User Centered System Design, New Perspectives on Computer Interaction, Hillsdale, Lawrence Erlbaum Associates, (1986), 31-61.
- [POIRIER-LEFEBVRE 93]  
F. Poirier et P. Lefebvre, Une interface multimodale de dialogue homme-machine sous Unix, Actes de Informatique'93, l'interface des mondes réels & virtuels, Montpellier, (Mars 1993), 161-170.
- [WELLNER 93]  
P. Wellner. "Interacting with paper on the DigitalDesk". Communications of the ACM, Vol.36 No.7, July 1993.