Essai sur l'Interaction Homme-Machine et son évolution

Joëlle Coutaz [1]

Introduction

Cet article est une réflexion personnelle sur l'Interaction Homme-Machine, domaine auquel j'ai contribué à petits pas depuis une bonne trentaine d'années. Il s'agit donc d'une analyse incomplète et nécessairement biaisée. Mes amis informaticiens, qu'ils soient académiques ou industriels, ont (trop) longtemps sous-estimé le rôle de l'Interaction Homme-Machine dans les processus de développement des logiciels. Cet article est une réponse à ce constat.

Posons quelques définitions pour préciser notre propos. Un système interactif est un système physico-numérique en interaction avec un (ou plusieurs) humain(s). *En interaction* implique l'existence d'actions réciproques et donc quelques influences mutuelles entre système et humains. On a coutume de structurer tout système interactif en deux grandes classes de fonctions : (1) l'ensemble des services propres à un domaine d'application : c'est le *noyau fonctionnel* du système ; (2) l'ensemble des fonctions et éléments matériels servant d'intermédiaires entre le noyau fonctionnel et l'humain : c'est *l'interface homme-machine* du système. Les éléments matériels de cette interface, comme l'écran et la souris, perceptibles et actionnables par l'humain, tiennent le rôle de *ressources d'interaction*. L'*Interaction Homme-Machine* désigne soit le phénomène [d'interaction], soit le domaine qui a pour objet d'étude ce phénomène.

1. Professeur émérite, Université Joseph Fourier, Laboratoire d'Informatique de Grenoble (LIG), UMR CNRS 5217. Email : joelle.coutaz@imag.fr

Dans ce qui suit, je démontre que l'Interaction Homme-Machine est un problème de conformité (deuxième section), que le contexte d'usage en est un élément clef (troisième section), puis j'analyserai l'évolution de l'Interaction Homme-Machine au cours de ces dernières cinquante années (quatrième section) avant de rappeler en synthèse les quelques messages que je considère importants.

Interaction Homme-Machine : une affaire de conformité

L'Interaction Homme-Machine (IHM) n'est pas seulement une affaire d'esthétisme, de coloriage ou de gesticulation à la *Minority Report*[2] ! Elle a, avant toute chose, *obligation de conformité*. En effet, l'IHM, en tant que discipline, a pour mission l'élaboration de théories, de modèles, de méthodes et de solutions techniques pour la conception et la mise en œuvre de systèmes interactifs *utiles*, *utilisables* et *plaisants*, et ceci *en toute circonstance*. Utilité, utilisabilité, plaisir et circonstance se mesurent, selon moi, en termes de conformité.

L'*utilité* d'un système interactif se caractérise par sa *conformité fonctionnelle*. Un système est fonctionnellement conforme si son noyau fonctionnel offre les services attendus par l'utilisateur cible (ni plus, ni moins), et si l'agencement de ces services respecte les raisonnements et procédés de l'utilisateur cible. Par exemple, si, pour connaître le prix d'un titre de transport, je dois réserver ma place, alors le système offre bien les bons services (« fournir un tarif » et « réserver une place »), mais ces services ne sont pas agencés de manière conforme à mes attentes puisque je me vois contrainte de faire une réservation pour atteindre mon objectif (connaître le prix). Ou encore, un utilisateur choisira tel système de chauffage pour faire des économies alors que pour un autre (voire pour ce même utilisateur, mais en des circonstances différentes), ce sera, non plus l'économie, mais le confort qui deviendra clef. Dès lors, le système de chauffage devra répondre à des exigences éventuellement contradictoires.

L'*utilisabilité* d'un système interactif se mesure en termes de *conformité interactionnelle*. Un système est conforme sur le plan interactionnel si son interface homme-machine présente les informations (ni plus, ni moins) qui permettent à l'utilisateur cible d'évaluer correctement l'état actuel du système, et d'identifier correctement le champ du possible pour atteindre ses objectifs, et de plus, si la représentation de ces informations est en adéquation avec les connaissances et les capacités sensori-motrices de cet utilisateur. Reprenons l'exemple du chauffage. Pour le premier utilisateur, l'économie réalisée est un concept central alors que pour le second, la température est pertinente. Dans le premier cas, l'interface homme-machine doit *rendre observable* l'économie réalisée alors que dans le second, ce sera la température.

2. Film de science fiction réalisé par Steven Spielberg en 2002 dans lequel l'acteur principal, Tom Cruise, explore un espace d'information projeté sur de grands écrans au moyen de gestes extrêmement sophistiqués pour l'époque.

Fournir la bonne information au bon moment est nécessaire mais pas suffisant. Il convient que cette information soit *représentée de manière compréhensible et accessible*. Dans notre exemple, exprimer la température en degrés Fahrenheit à destination d'un utilisateur de culture française n'est pas judicieux ou pire, ne pas indiquer l'unité de mesure peut conduire à une interprétation erronée ! En outre, si l'utilisateur est malvoyant et que la température est représentée par un dessin (on parle alors de modalité visuelle), alors cette représentation doit être accompagnée d'un texte sémantiquement équivalent en sorte qu'un lecteur d'écran (*screen reader*) lise ce texte à voix haute [3].

La *qualité hédonique* d'un système interactif est plus délicate à cerner. Elle fait l'objet de recherches actives en relation notamment avec l'informatique affective initiée par Rosalind Picard [39]. C'est à ce niveau que l'esthétisme évoqué plus haut entre en jeu, mais pas seulement ! La communauté tend à utiliser un terme valise, *the user experience*, pour désigner un vécu subjectif, le sens et la valeur qu'un individu attache à un système, mais aussi… l'utilité et l'utilisabilité perçues.

L'exemple du système de chauffage montre que les requis fonctionnels, que les connaissances, capacités cognitives et sensori-motrices de l'utilisateur cible, ainsi que les ressources d'interaction, ne sont ni immuables, ni totalement prévisibles. Si l'on pose comme principe que les conformités fonctionnelle et interactionnelle doivent être satisfaites en toute circonstance, il convient dès lors de se pencher sur la question de l'adaptation des systèmes en sorte de garantir la *conformité contextuelle*. Un système interactif est conforme sur le plan contextuel si les conformités fonctionnelle et interactionnelle sont assurées quel que soit le contexte d'usage.

Conformité contextuelle en Interaction Homme-Machine : un défi de tous les instants

Contexte : encore un terme surchargé, y compris dans la communauté de recherche en informatique [19, 21]. Reprenons à notre compte cette phrase de P. Brézillon, « il n'y a pas de contexte sans contexte ! » [9]. Autrement dit, le contexte n'existe pas en tant que tel. Il émerge, ou se définit, pour une finalité précise. La finalité qui nous concerne ici est l'adaptation du système interactif (donc, de son noyau fonctionnel et/ou de son interface homme-machine) en sorte que les conformités fonctionnelle et interactionnelle, de même la valeur, soient garanties. Pour cette raison, nous parlons ici, non pas de contexte-tout-court, mais de *contexte d'usage* que nous structurons en trois sous-espaces interdépendants : le (ou les) utilisateur(s) du système interactif, la (ou les) plate(s)-forme(s) d'exécution et notamment leurs ressources d'interaction, le (ou les) environnements physique(s) et social(sociaux) dans lequel (lesquels) a lieu l'interaction [12].

3. À ce titre, les recommandations WCAG (Web Content Accessibility Guidelines) élaborées par le W3C, même si elles peuvent paraître verbeuses et complexes, doivent être considérées avec attention.

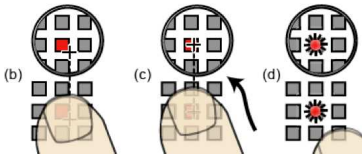


FIGURE 1. Shift [47]. (b) Le doigt masque la cible. La partie masquée est décalée et la position actuelle du curseur est montrée. (c) Le curseur est déplacé en couplage étroit avec le mouvement du doigt maintenu sur la surface. (d) Sélection de la cible au relâcher du doigt.

L'importance du contexte d'usage est reconnue depuis longtemps en conception des systèmes interactifs. La conception contextuelle (*contextual design*) de Beyer et Holtzblatt, fondée sur la collecte et l'organisation de données de terrain (*contextual inquiry*), en est un exemple type [7]. Les modèles de l'Action Située (*Situated Action*) [45], la Théorie de l'Activité [5], la Cognition Distribuée (*Distributed Cognition*) [26] ou encore la conception par scénarios [16] visent autre chose que la décomposition stricte de GOMS [15] en tâches et sous-tâches (buts et sous-buts) en considérant le contexte comme élément directeur.

Si le contexte d'usage est utilisé depuis longtemps en conception des systèmes interactifs, en pratique, il est essentiellement exploité en phase amont du processus de développement. Tout ce qui relève du contexte d'usage se dilue progressivement au cours du processus de développement et les conformités fonctionnelle et interactionnelle sont au mieux assurées pour un ensemble de situations prévues à l'avance. Avec l'explosion de la diversité des plates-formes d'exécution et notamment des smartphones, la communauté a développé, pour des tâches récurrentes, des techniques d'interaction appropriées. Illustrons notre propos avec l'exemple de la tâche suivante : indiquer au système le choix d'une valeur parmi N. Le concept de menu a été inventé pour faciliter cette tâche en servant d'extension à la mémoire à court terme (il est plus facile pour l'humain de reconnaître 1 valeur parmi N que de la reconstituer *ex nihilo* [34].)

La souris, en couplage étroit avec les écrans à points, a rendu possible la mise en œuvre du menu graphique linéaire. S'appuyant sur la loi de Fitts (expliquée et exploitée dans [15]), on invente le menu circulaire qui permet de réduire le temps de sélection d'un élément sous réserve que le menu comporte au plus huit éléments : au-delà, la taille des cibles diminuant, le temps de sélection augmente, de même, le risque d'erreur, surtout si la sélection doit se faire au doigt sur un écran tactile. Pour éviter de masquer une cible trop petite, Shneiderman puis Vogel & Baudisch introduisent la technique Shift qui décale légèrement la cible du doigt [47] (Figure 1) –-



FIGURE 2. LucidTouch : écran translucide [48].

problème qui ne se pose pas si l'on dispose d'un écran translucide comme LucidTouch [48] (Figure 2).

La petitesse des écrans des smartphones introduit de nouvelles contraintes. Alors, avec le menu Wavelet, on sait afficher des listes hiérarchiques sur une petite surface [24] (Figure 3). Inversement, les capteurs de mouvements offrent de nouvelles opportunités d'innovation fondées sur le geste : un menu linéaire se métamorphose en feuillets 3D lorsque l'utilisateur incline le téléphone (effet ludique). Sur les tables interactives multipoints, chaque doigt de la main vient servir de point d'entrée au menu MTM [2], comme les accords que l'on forme avec les touches d'un piano [6] : on exploite le parallélisme combinatoire des facultés humaines. Si la projection d'informations a lieu sur un mur, alors l'ombre du doigt, plutôt que le doigt, permet de désigner la cible hors de portée [44], ou bien un pico-projecteur sert à la déplacer vers soi pour la manipuler plus confortablement [14], ou encore, comme avec SkinPut [27] (Figure 10(c)), le menu est projeté sur l'avant-bras et l'impact du doigt sur l'avant-bras est détecté par un brassard équipé de capteurs bioacoustiques.

Voilà comment, au fil des années, un certain pan de la recherche en IHM procède par incréments, combinant théories psycho-cognitives, contraintes et opportunités techniques pour « augmenter la bande passante » entre le système et l'utilisateur. Cette recherche, centrée sur l'invention de nouvelles techniques d'interaction, vise à résoudre des micro-problèmes d'interaction précis pour des situations d'usage précises, par exemple le pointage de cible dans une scène graphique pour un smartphone à écran tactile. J'appelle cette recherche *IHM à pointe fine* en complément de l'*IHM systémique* qui, elle, envisage un système interactif dans sa globalité pour des usages dans le monde réel avec sa diversité et ses aléas. Les méthodes de conception, les architectures logicielles de systèmes interactifs, les boîtes à outils et les générateurs d'interfaces homme-machine, de même l'étude de la plasticité des interfaces homme-machine, relèvent de l'Interaction Homme-Machine systémique.

La *plasticité* de l'interface homme-machine d'un système interactif dénote la capacité d'adaptation de cette interface au contexte d'usage pour en préserver la conformité contextuelle tout en accordant à l'utilisateur les moyens de contrôle adéquats.



FIGURE 3. Déclinaison du menu. De gauche à droite et de haut en bas : le menu Wavelet sur téléphone mobile (le menu hiérarchique se déroule comme des vaguelettes imbriquées en déplaçant le doigt du centre vers la périphérie) ; la sélection d'un élément du menu par relâcher du doigt donne accès au résultat ou bien tant que le doigt est posé sur le menu, le résultat apparaît sur le menu en transparence donnant ainsi une idée de la nature des résultats [24] (principe du *feedforward*) ; au centre, le Menu polymorphe 2D/3D ; en bas, MTM (Menu MultiTouch) pour table interactive multipoints [2], Shadow Reaching : sélection de cible avec son ombre [44].

On trouvera dans [18] la description de l'espace problème de cette propriété, de même les moyens de la mettre en œuvre techniquement par l'utilisation conjointe, à l'exécution, de transformations de modèles (approche IDM – Ingénierie Dirigée par les Modèles) et de recrutement et/ou de (re)composition dynamiques de composants orientés services [3, 22]. Si, le « Cameleon Framework » [12] sert aujourd'hui de cadre conceptuel canonique pour la plasticité des interfaces homme-machine, les outils à grande portée ne sont que des prototypes de laboratoire à durée de vie limitée.

Le mouvement *Responsive Web Design* traite de l'adaptation dynamique de pages web à la plate-forme d'exécution, mais ne couvre pour l'heure, avec les *CSS media queries*, qu'un remodelage cosmétique de surface, non pas une adaptation profonde de niveau tâche, encore moins la redistribution dynamique de l'interface homme-machine sur les ressources d'interaction de l'espace. De plus, ne sont considérées que des IHM centralisées !

Et que dire de la reconfiguration dynamique des services du noyau fonctionnel ? Ce problème, qui fait l'objet de recherches extrêmement actives en Génie logiciel et systèmes répartis, n'est pas résolu dans toute sa généralité [13, 40, 49]. En IHM, la plasticité relève du même problème, mais en plus *elle se voit* ! Le fait que « ça se voit » implique que les états successifs du processus d'adaptation doivent être observables. J'appelle *interface homme-machine de transition*, les éléments interactionnels du système qui véhiculent cette observabilité. Les animations graphiques entre deux états stables, par exemple le tracé rapide de rectangles de taille variable lors de la fermeture d'une fenêtre, et plus généralement les techniques de *morphing*, sont des IHM de transition classiques. Encore faut-il que ces animations respectent les capacités sensorielles de l'utilisateur – ni trop rapides, ni trop lentes, pour que l'évaluation du changement d'état puisse se faire facilement et correctement. Aujourd'hui, nous ne disposons pas de patrons d'IHM qui traduisent visuellement la migration d'éléments graphiques en interaction multi-surfaces [4]. La disparition progressive d'un composant de la surface source (par exemple le panneau de contrôle du visualisateur de diapositives de la Figure 4) suivie de sa réapparition progressive sur la surface destinataire est une solution à condition que les deux surfaces soient dans le champ de vision de l'utilisateur. Un rendu multimodal alliant les modalités son-3D et graphique pourrait apporter une réponse, mais son efficacité reste à démontrer.

En somme, la conformité contextuelle d'un système interactif qui implique conformité fonctionnelle et conformité interactionnelle quel que soit le contexte d'usage (utilisateur, plate-forme, environnement physique et social), est un combat d'envergure qui n'est pas encore gagné.

Voilà, en quelques pages, mon analyse de l'état des lieux de la recherche en Interaction Homme-Machine conduites par deux courants complémentaires : l'IHM à pointe fine et l'IHM systémique. Considérons maintenant le chemin parcouru en nous plaçant cinquante ans en arrière.

L'Interaction Homme-Machine en trois grandes étapes

Je propose d'analyser l'évolution de l'Interaction-Homme Machine en prenant comme élément directeur la nature des ressources qui se sont successivement trouvées au cœur des préoccupations. Ainsi, nous comptons trois étapes : l'ordinateur en tant que ressource critique, puis l'individu comme sujet de toutes les attentions et aujourd'hui, le monde physique et social en relation avec le monde numérique.



(a) (b) (c)

FIGURE 4. Visualisateur de diapositives CamNote [20]. Exemple d'IHM de transition qui rend observable la migration du panneau de contrôle entre un écran de PC projeté sur le mur et le PDA. (a) IHM centralisée sur PC avant l'arrivée du PDA. (b) Le panneau de contrôle visible sur le mur glisse vers la diapositive pour s'y fondre progressivement. (c) Le panneau de contrôle a disparu du mur. Il réapparaît progressivement sur le PDA après avoir été remodelé sous forme de gros boutons pour faciliter la sélection en situation de mobilité.

Près de cinquante ans en arrière : l'ordinateur, une ressource critique isolée

Cinquante ans en arrière : l'ordinateur est unique, cher et encombrant. Il vit sous cloche dans de très grands centres de calcul. La figure 5 en montre un exemple. L'utilisateur n'accède pas directement à cette machine, ce privilège étant réservé à des opérateurs spécialisés. Les programmes sont saisis sur cartes perforées, les résultats sont imprimés sur des accordéons de feuilles de papier et les bandes magnétiques et les disques amovibles servent au stockage des données. Au mieux, la mémoire centrale frise le mégaoctet. Absence de réseau, puisque la commutation de paquets est un sujet de recherche d'avant-garde [4].

L'utilisateur est un programmeur rompu au calcul scientifique, aux statistiques et aux applications de gestion (fiches de paie). Il est attentif à la consommation des ressources qu'il précise explicitement dans un langage de commande dédié (JCL ou *Job Control Language*). Ainsi, un programme qui « bouclerait » ou qui demanderait plus de mémoire que prévu, ou encore qui imprimerait plus de pages que prévu, serait automatiquement terminé par le système d'exploitation. Le programmeur est responsable de sa consommation de ressources de calcul. Tout l'art tient à sa capacité à produire des programmes corrects « du premier coup » (il a droit à une exécution par jour !), y compris faire de « l'overlay à la main » en sorte que le programme « tienne » dans la mémoire centrale disponible — le concept de mémoire virtuelle est

4. On doit à Louis Pouzin, alors chercheur à l'IRIA (pas encore l'INRIA), et à son équipe, la première mise en œuvre mondiale du concept de datagramme dans le cadre du projet Cyclades. La première démonstration a eu lieu en 1973, le réseau comprenant alors trois nœuds situés respectivement à l'IRIA, à la CII et à l'IMAG à Grenoble.



(a) (b)

FIGURE 5. (a) Ordinateur pour tous, ici, l'IBM 360, fonctionnant soit en traitement par lots (essentiellement la nuit), soit en temps partagé (le jour). (b) Un bac de cartes constituant un programme bien tangible ! Les procédures du programme sont mises en évidence par des marques et des noms inscrits au crayon feutre.

encore un sujet de recherche. Mais, comme le montre la figure 5, les programmes et leurs résultats sont tangibles : l'empreinte d'un programme et sa modularité peuvent se mesurer à l'œil !

Avec l'apparition des systèmes à temps partagé, les cartes perforées disparaissent au profit des terminaux personnels (les célèbres « télétypes » à papier) remplacés progressivement par les écrans alphanumériques (ou *glass teletypes*) qui, comme leur nom l'indique, ne sont pas encore des écrans à points. Si nous gagnons en performance, nous perdons la matérialité de nos programmes. L'optimisation des ressources et leur « virtualisation » (que l'on retrouve aujourd'hui avec le *cloud computing* et les systèmes embarqués sur capteurs) sont restées les moteurs de l'informatique jusqu'à ce que quelques chercheurs (nord-américains et nord-européens pour la plupart) portent leur attention sur la ressource humaine. En effet, le coût du personnel d'entreprise est sur le point de l'emporter sur celui des machines.

Il y a trente ans : l'utilisateur au centre des préoccupations

1983 est l'année de la première édition de la conférence CHI (*Computer Human Interaction*) de l'ACM[5]. Cette conférence, concomitante du déploiement des calculateurs personnels, marque le début d'une quête militante pour le développement d'applications utiles et utilisables pour un utilisateur cible. L'utilisateur n'est plus le programmeur de l'époque précédente mais un non-spécialiste de l'informatique

5. CHI'83 fait suite au tout premier atelier sur le sujet à Gaithersburg en mars 1982, intitulé *Human Factors in Computer Systems*. En France, 1989 est la première édition de la conférence IHM tenue à l'INRIA Sophia-Antipolis.

qu'il convient de soutenir dans ses activités professionnelles. Nous passons de l'utilisateur programmeur au bureauticien. À cet égard, citons le buroviseur [6] du projet Kayak lancé par l'I(N)RIA en 1981. L'*efficacité humaine* est l'élément directeur de la recherche. On ne parle pas encore de *user experience*, ni de qualité hédonique ou de valeur, mais cette période est marquée par la pose des fondements de l'IHM que ce soit en méthodes, théories et techniques logicielles.

Concernant les méthodes, citons la conception participative et contextuelle [7], la conception itérative (dont l'esprit s'accommode bien des méthodes « agiles » du génie logiciel), de même la conception dirigée par les scénarios [41]. Ces méthodes centrées utilisateurs ont donné naissance à de nombreux formalismes tels CLG [15], TAG [38] et ETAG [46], UAN [28], CTT [37], KMAD [11] pour modéliser les plans mentaux des utilisateurs cibles sous forme de modèles de tâches (une structure arborescente de buts et de sous-buts reliés par des opérateurs de composition ou de relations temporelles). Un tel modèle, bien plus complet qu'un *Use Case* UML, spécifie les requis fonctionnels et leurs enchaînements du point de vue de l'utilisateur. Produire un modèle de tâche, pour peu qu'il soit transformé automatiquement en code exécutable, est une bonne façon de garantir la conformité fonctionnelle dont il a été question plus haut.

Du côté des théories, au-delà du modèle du processeur humain [15], le principe de correspondance directe de Norman entre variable psychologique et objet informatique, entre état interne du système et sa représentation et sa manipulation, font aujourd'hui partie (ou devrait faire partie) du bagage intellectuel de base de tout informaticien [35]. L'application stricte de ce principe participe à la conformité interactionnelle et notamment à ce que l'observabilité des concepts pertinents soit assurée — cf. exemples de l'économie d'énergie et de la température cités plus haut.

À l'époque, deux types de représentation (et de recherches), aujourd'hui reconnues comme modalités d'interaction complémentaires, s'affrontaient : les représentations langagières (langue naturelle ou langage artificiel comme les *shells*) et les représentations fondées sur la *métaphore du monde réel*, telle la célèbre métaphore du bureau qui prévaut encore aujourd'hui (voir Figure 6). On recherchait, avec les moyens techniques de l'époque, à recréer la matérialité du monde réel que nous avions perdue avec le temps partagé de l'étape précédente.

Du côté technique, le paradigme d'interaction WIMP (*Window, Icon, Menu, Pointing*) et ses commandes génériques (couper-coller, imprimer, glisser-déposer, etc.), véhiculés par les boîtes à outils graphiques natives et par les patrons de conception d'IHM, sont le résultat de ces théories et principes. En outre, les informaticiens impliqués dans les développements des interfaces homme-machine ont vite compris

6. Le buroviseur a été présenté au SICOB 83 à Paris. On sera étonné par les techniques déjà employées il y a plus de trente ans : voir `http://www.sicob.tv/1983-Le-magazine-Temps-Reel-decrypte-l-actualite-du-salon_a94.html`.

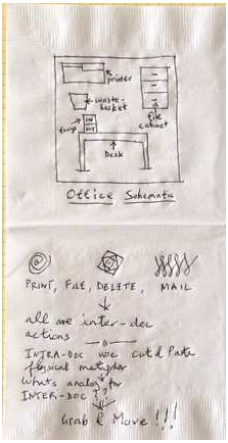


FIGURE 6. À la recherche d'une représentation graphique de la métaphore de bureau, une idée déjà dans l'air au milieu des années 1970 : esquisse que Tim Mott a produite vers la fin des années 1970 (tirée de [32], p. 52). Notons la présence des premières commandes génériques : « Print, File, Delete, Mail, Cut and Paste, Grab and Move ».

que leurs programmes devaient être facilement modifiables — nécessité de développements itératifs, et qu'une « bonne » modularité fondée sur la séparation des aspects était clef. C'est ainsi que sont nés MVC [29] (aux USA), et PAC [17] (en France) bien avant que l'architecture logicielle soit un sujet de recherche. De même, les générateurs d'IHM de ces années-là appliquaient déjà les principes de l'IDM sans le savoir.

Du côté de l'évaluation, ergonomes et psychologues ont transmis (tant bien que mal) aux informaticiens leurs méthodes d'élaboration de protocoles expérimentaux pour mesurer les performances humaines, le plus souvent en termes de temps d'exécution et de nombre d'erreurs. Il convient de souligner que la définition de ces protocoles est délicate car, comme en chimie et en physique, on a tôt fait de biaiser les résultats. De plus, le dépouillement et l'analyse des données sont particulièrement chronophages. Et pourtant, ces tests valent (presque) autant qu'une campagne de tests du Génie Logiciel !

En parallèle, Internet et les réseaux sans fil, le *web* (qui fêta ses vingt ans en 2010), de même les navigateurs — utiles et utilisables par (presque) tous — changent la donne. De la machine individuelle isolée, nous passons à l'ère du « tout instantanément connecté » de la troisième période.



(a)



(b)



(c)



(d)

FIGURE 7. Exemples de services et de dispositifs. (a) Projection d'ambiance sur le sol. b) Le « napperon-mémoire » [25] qui trace le déplacement d'objets chez soi : un halo lumineux se forme progressivement autour de l'objet, puis disparaît progressivement lorsque l'objet est placé ailleurs. (c) Application du Lily Pad Arduino : fabrication personnelle d'une veste augmentée d'un clignotant pour indiquer un changement de direction [10]. (d) Les siftables composables à façon pour construire de nouvelles applications [31].

Depuis une dizaine d'années : union des mondes physique, social et numérique

Par opposition à l'informatique de la deuxième période, dont l'ordinateur de bureau est l'archétype, la nouveauté technique tient aux capacités de mobilité et d'intégration des systèmes numériques dans le milieu physique au point de s'y confondre, et ceci de manière spontanée, à de multiples échelles, de la planète au micro, voire nano-objet. Cette mobilité et cette fusion rendues possibles par la miniaturisation et la puissance des composants électroniques, par l'omniprésence des réseaux, et par la chute des coûts de production, permettent d'entrevoir la composition opportuniste de dispositifs et de services de toutes sortes au-dessus d'une infrastructure à granularité et à géométrie variables, dotés de facultés de capture, d'action, de traitement, de communication et d'interaction. La figure 7 illustre en images cette tendance.

Les exemples de la figure 7 appellent trois remarques : *polymorphisme* de l'ordinateur ; *tissage*, au sens propre comme au figuré, du numérique dans nos activités ; de *l'utile au futile*. Autrement dit, le *monde physique est devenu une ressource façonnable et (re)constructible par l'individu*, pas seulement pour être plus efficace,



FIGURE 8. Toute phrase KISS comprend une partie condition « Quand » et une partie action « Je veux ». L'écriture de phrases se fait via des menus qui proposent des options syntaxiquement et sémantiquement correctes : les menus sont alimentés par une base de connaissances. Dans cet exemple, l'état actuel de la phrase est le suivant : « Quand il est 21h, je veux éteindre la lumière de la salle de bain (et écouter [quelque chose] si l'utilisateur sélectionne "écouter") ». Le bouton « + » permet d'insérer des conditions ou des actions. Les phrases KISS sont traduites en Aspect d'Assemblage pour interprétation par WCOMP [22].

mais pour améliorer la qualité de vie, le bien-être, le plaisir... l'expérience. On parle aussi de « funologie » [8]. L'utilisateur n'est plus nécessairement un sujet condamné à consommer les applications imposées par l'offre du marché. Il a maintenant la possibilité d'être un acteur, tel le bricoleur du dimanche qui construit et améliore son espace de vie à partir de composants disponibles sur étagère.

Le faire soi-même (*DiY — Do it Yourself*) reçoit un franc succès dans tous les domaines. Chez nous, la diffusion des imprimantes 3D et des *Fab Labs* ouvrent à tous de nouvelles perspectives. De mon point de vue, il est temps de *redonner le pouvoir à l'utilisateur* et pour cela, lui fournir les environnements de développement (*End-User Development Environment* – EUDE) adaptés [33]. Les résultats de KISS (*Knit your Ideas into Smart Spaces*), un EUDE pour maison intelligente, nous encouragent dans cette voie [23].

KISS propose un langage de programmation de type déclaratif orienté règles, avec potentiel d'égale opportunité syntaxique entre langue française pseudonaturelle (LPN) et langage visuel iconique. Comme le montre la figure 8, la technique d'interaction de construction des programmes LPN s'appuie sur l'utilisation de menus dont les options sont calculées dynamiquement assurant ainsi la découverte progressive du langage ainsi que l'extensibilité de la grammaire, de même la correction syntaxique des programmes. La mise au point peut se pratiquer, au choix, dans le monde physique ou dans un monde dual numérique (cf. Figure 9). L'évaluation de KISS dans Domus, un habitat intelligent d'expérimentation, montre que les utilisateurs parviennent à programmer des scénarios réalistes de la vie quotidienne. Je retiens aussi que la programmation et les processus de développement associés sont en train d'échapper aux professionnels de l'informatique. Les *mashups* n'en sont qu'à leur début, mais les outils comme Natural Mash-Up qui utilise un LPN comme langage de spécification, émergent rapidement [1].

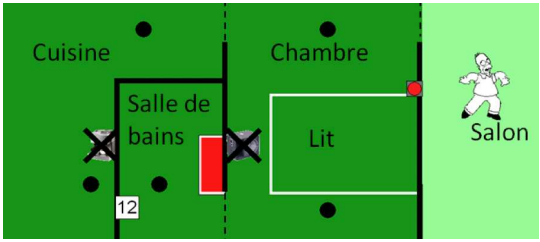
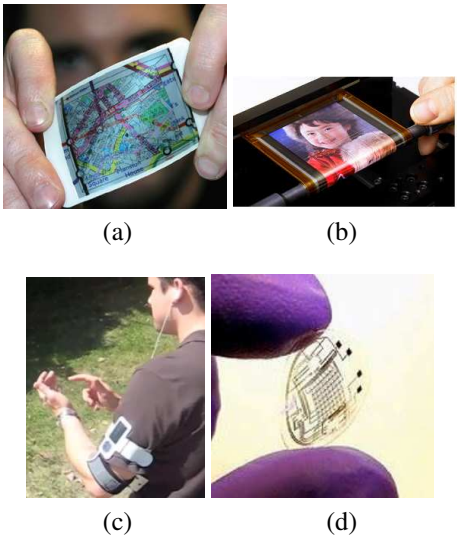


FIGURE 9. Le metteur au point est une reproduction 2D duale de l'environnement réel. Ici, l'appartement de la plate-forme d'expérimentation Domus. Les effets de l'exécution d'un programme KISS se « voient » dans cette représentation. Le personnage « Homer » représente la position actuelle de l'utilisateur ; les points noirs, des lampes éteintes ; les traits gris en bas de la figure, les volets en position « fermé » ; et les croix sur les icônes représentatives de la cafetière et de la radio indiquent que ces dispositifs sont en arrêt.

Remarquons que cette capacité individuelle de création se voit démultipliée par un nouveau phénomène : la participation sociale [30] rendue possible par le *web*, premier système universel. En vérité, la dimension sociale est étudiée en lien avec l'informatique depuis la fin des années 1980[7]. L'objectif est à cette époque d'élaborer des modèles, théories et systèmes numériques, dits collecticiels, susceptibles d'améliorer les activités de groupe en termes de production, de coordination et de communication. Avec le *web*, le changement d'échelle suscite de nouveaux usages. Tout individu, toute collectivité et toute communauté, peut collecter des informations, les relier, puis produire de nouvelles informations et les offrir à son tour à l'humanité. C'est ce que Shneiderman appelle le *collect-relate-create-donate mantra* [43]. Wikipedia est le premier exemple réussi de construction collective de connaissances encyclopédiques. Il en est d'autres comme *Google Image Labeler* pour l'indexation des images et *TopCoder* pour la production sociale de programmes. Les magasins numériques de logiciels, initiés par l'*App Store* d'Apple, entraînent des changements dans les processus de développement des logiciels et, de là, l'émergence de nouveaux modèles (et opportunités) économiques.

En revanche, il nous faut survivre aux avalanches d'informations alors que la bande passante humaine, elle, reste constante. Il y a là un véritable goulot d'étranglement qui, au-delà des moteurs de recherche, nécessite l'invention de nouvelles techniques d'interaction. Dans cette direction, l'interaction gestuelle et les centrales

7. La première conférence ACM sur le sujet, *Computer Supported Collaborative Work* (CSCW), a eu lieu en 1987.



(a) (b)

(c) (d)

FIGURE 10. Exemples de nouveaux dispositifs illustrant l'intégration de connaissances STIC, des nanotechnologies au logiciel. (a) Une maquette du Gummi (Sony), un ordinateur déformable : la navigation dans l'espace d'information se fait en courbant la surface [43]. (b) Un écran OLED enroulable (présenté par Sony lors du SID, Society for Information Display, 2010). (c) Skinput appliqué au baladeur musical : les frappes du doigt sur la surface de la peau sont captées par un brassard muni de capteurs [28]. (d) Une lentille souple dans laquelle est imprimé un circuit pour affichage direct sur la surface de l'œil [37].

d'inertie pour téléphones mobiles, l'interaction corporelle et la vision par ordinateur avec reconstruction 3D temps réel pour grand public (tel Kinect [8] de MicroSoft), l'interaction musculaire, les écrans transparents multipoints et déformables figurent parmi les sujets particulièrement actifs (voir Figure 10). Ces exemples démontrent, s'il en est encore besoin, que l'innovation passe par un rapprochement inédit de spécialistes intra-STIC (sciences et technologies de l'information et de la communication) et entre STIC et SHS (sciences humaines et sociales), des nanotechnologies aux logiciels, à tout niveau d'abstraction, et orthogonalement de l'individu aux groupes sociaux.

8. Relevons, à Noël 2010, le slogan publicitaire de vente de Kinect affiché partout dans les couloirs du métro parisien : « c'est vous la manette ! », qui assimile l'homme à la célèbre manette Wii de Nintendo.

Synthèse

Les messages à retenir de cet essai sont les suivants :

(1) l'IHM, c'est bien plus que de la peinture ! C'est une affaire de conformités fonctionnelle, interactionnelle et contextuelle, dont l'impact concerne toutes les couches d'un système interactif (pensez à la fonction « défaire » : son impact dépasse largement la cosmétique de surface). La conformité contextuelle est un combat d'envergure qui n'est pas encore gagné.

(2) Au cours de ces cinquante dernières années, l'IHM est passée du statut « d'existence implicite » (il y a toujours eu des commandes pour contrôler un ordinateur, mais sans être le résultat d'une activité conception), puis, dans les années 80, l'IHM a pris le statut de « recherche marginale » en informatique (de trop nombreux systèmes interactifs ont été produits en considérant l'IHM après coup), pour enfin être reconnue comme discipline depuis une dizaine d'années (et encore !).

(3) La recherche en IHM est organisée en deux courants complémentaires : l'IHM à pointe fine qui vise à résoudre des problèmes précis d'interaction et l'IHM systémique qui étudie des problèmes de portée globale.

(4) Nous entrons dans une ère de changements radicaux avec, devant nous, des défis tels que STIC et SHS ne peuvent plus s'affranchir d'un rapprochement sincère et véritable.

Références

[1] S. Aghaee, C. Pautasso, Guidelines for Efficient and Effective End-user Development of Mashups, End-User Development, 4th International Symposium, IS-EUD 2013, 260-265, LNCS 7897, Springer, 2013 .

[2] G. Bailly, A. Demeure, L. Nigay, E. Lecolinet, MultiTouch Menu, IHM 2008, 20ème conférence francophone sur l'Interaction Homme-Machine, ACM Press, 2008.

[3] L. Balme, A., Demeure, N. Barralon, J. Coutaz, G. Calvary, CAMELEON-RT : A Software Architecture Reference Model for Distributed, Migratable, and Plastic User Interfaces, *Lecture Notes in Computer Science*, Vol. 3295, 2004. In P. Markopoulos, B. Eggen, E. Aarts et al. (Eds), Ambient intelligence : Second European Symposium, EUSAI 2004. Springer-Verlag Heidelberg (Publisher), ISBN : 3-540-23721-6, 291-302, 2004.

[4] R. Balakrishnan, P. Baudisch, Special Issue on Ubiquitous Multi-Display Environments, *Human-Computer Interaction*, Vol. 24, Taylor and Francis, 2009.

[5] J.E. Bardram, Plans as Situated Action : An Activity Theory Approach to Workflow Systems, Proceedings of ECSCW'97 Conference, Lancaster UK, 1997.

[6] O. Bau, *Interaction streams : helping users learn, execute and remember expressive interaction grammars*, Thèse de doctorat, Université Paris-Sud, 2010.

[7] H. Beyer H, K. Holtzblatt, *Contextual Design : Defining Customer-centered Systems*, Morgan Kaufman, 1998.

[8] M.A. Blythe, K. Overbeeke, A. Monk, P. Wright, From Usability to Enjoyment, *Human Computer Interaction Series*, vol. 3, Springer, New York, 2003.

[9] P. Brézillon, *Expliciter le contexte dans les objets communicants. Les Objets Communicants.* Hermes Sciences Editions, Lavoisier, Chapitre 21, 295-303, 2002.

[10] L. Buechley, M. Eisenberg, J. Catchen, A. Crockett, The LilyPad Arduino : Using Computational Textiles to Investigate Engagement, Aesthetics, and Diversity, Proceedings of the SIGCHI conference (CHI 2008), 423—432, 2008.

[11] S. Caffiau, D. Scapin, P. Girard, M. Baron, F. Jambon, Increasing the expressive power of task analysis : systematic comparison and empirical assessment of tool-supported task models, *Interacting with Computers*, Elsevier Science B.V, 22(6), 569-593, 2010.

[12] G. Calvary, J. Coutaz, D. Thevenin, Q. Limbourg, N. Souchon, L. Bouillon, J. Vanderdonckt, A Unifying Reference Framework for Multi-target User Interfaces, *Interacting with Computers*, Elsevier Science B.V., 15(3), 289—308, 2003.

[13] G. Canfora, M. Di Penta, P. Lombardi, M.L. Villani, Dynamic Composition of Web Applications in Human centered Processes, IEEE PESOS'09, 2009.

[14] X. Cao, R. Balakrishnan, Interacting with dynamically defined information spaces using a handheld projector and a pen, Proceedings of UIST 2006, ACM Symposium on User Interface Software and Technology, ACM Press, 225—234.

[15] S.K. Card, T.P. Moran, A. Newell, *The Psychology of Human-Computer Interaction*, Lawrence Erelbaum Associates, 1983.

[16] J.M. Caroll, Making Use, *Scenario-based design of Human-Computer Interactions*, MIT Press, 2000.

[17] J. Coutaz, PAC, An Implemention Model for Dialog Design, Proceedings of Interact'87, IOS Press, 431-436, 1987.

[18] J. Coutaz, G. Calvary, Chapitre 9 - *Systèmes interactifs et adaptation centrée-utilisateur : la plasticité des Interfaces Homme-Machine, Informatique et Intelligence Ambiante*, Hermes Science Pub., 2012.

[19] J. Coutaz, J.L. Crowley, S. Dobson, D. Garlan, Context is key, *Communications of the ACM*, ACM Publ., 48(3), 49-53, 2005.

[20] A. Demeure, L. Balme, G. Calvary, CamNote : A plastic slides viewer, In Proc. Plastic Services for Mobile Devices (PSMD) workshop held in conjunction with INTERACT'05, 2005.

[21] A.K. Dey, Understanding and using Context, *Journal of Personal and Ubiquitous Computing*, Springer London, Vol 5, 4-7, 2001.

[22] N. Ferry, G. Hourdin, S. Lavirotte, G. Rey, J.-Y. Tigli, M. Riveill, Models at Runtime : Service for Device Composition and Adaptation, in 4th International Workshop Models@run.time, Models 2009 (MRT'09), 2009.

[23] E. Fontaine, *Programmation d'espaces intelligent par l'utilisateur final*, Thèse de doctorat de l'Université de Grenoble, 2012.

[24] J. Francone, G. Bailly, N. Mandran, L. Nigay, Wavelet Menus on Handheld Devices : Stacking Metaphor for Novice Mode and Eyes-Free Selection for Expert Mode, Proc. of the Conference on Advanced Visual Interfaces (AVI), ACM Pub., 2010.

[25] W. Gaver, The History Tablecloth ; Illuminating Domestic Activity, DIS'06 Proceedings of the 6th conference on Designing Interactive systems, 199-208, ACM Pub., 2006.

[26] C.A. Halverson, Distributed Cognition as a theoretical framework for HCI : Don't throw the Baby out with the bathwater - the importance of the cursor in Air Traffic Control. Tech Report No. 94-03, Dept. of Cognitive Science, University of California, San Diego, 1994.

[27] C. Harrison, T. Desney, D. Morris, Skinput : Appropriating the Body as an Input Surface. In Proceedings of CHI'10, the 28th international conference on human factors in computing systems, ACM, 453–462, 2010.

[28] R. Hartson, P. Gray, Temporal Aspects of Tasks in the User Action Notation, *Human Computer Interaction*, vol. 7, 1—45, 1992.

[29] G.E. Krasner, S.T. Pope, A cookbook for using the Model-View-Controller user interface paradigm in Smalltalk-80. *Journal of Object Oriented Programming (JOOP)*, 1(3), 26—49, 1988.

[30] R. Kraut R., M.L. Maher, J. Olson, T. Malone, P. Pirolli, J.C. Thomas, Scientific Foundations : a case for Technology-Mediated Social Participation Theory, *IEEE Computer*, 22—28, 2010.

[31] D. Merrill, J. Kalanithi, P. Maes, Siftables : Towards Sensor Network User Interfaces, Proceedings of the First International Conference on Tangible and Embedded Interaction (TEI'07), 2007.

[32] B. Moggridge, *Designing Interactions*, The MIT Press, Cambridge, 2006.

[33] B. Nardi, A Small Matter of Programming, *Perspectives on End User Computing*. The MIT Press, Cambridge, 1995.

[34] D.A. Norman, D.G. Bobrow, Descriptions : An intermediate stage in memory retrieval, *Cognitive Psychology*, 11, 107-123, 1979.

[35] D.A. Norman, S.W. Draper, *User Centered System Design : New Perspectives on Human-Computer Interaction*. Erlbaum Associates, Hillsdale, NJ, 1986.

[36] B. Parviz, Augmented Reality in a Contact Lens, IEEE Spectrum, septembre 2009.

[37] F. Paternò, Concur Task Trees : An engineered notation for task models. In D. Diaper, & N. Stanton, (Eds.), *The Handbook of Task Analysis for Human-Computer Interaction*, Lawrence Erlbaum Associates, 483-503, 2003.

[38] S. Payne, T. Green, Task-Actions Grammars : A Model of the Mental Representation of Task Languages, *Human-Computer Interaction*, 2, 93-133, 1986.

[39] R.W. Picard, *Affective Computing*, MIT Press, 1997.

[40] S. Pietschmann, M. Voigt, K. MeiBner, Dynamic Composition of Service-Oriented Web User Interfaces, proc. of the 4th International Conf. on Internet and Web Applications and Services, ICIW 2009, IEEE CPS, ISBN 9780769536132, 2009.

[41] M.B. Rosson, J.M. Carroll, *Usability Engineering, Scenario-based Development of Human Computer Interaction*, Morgan Kaufmann, San Francisco, 2002.

[42] C. Schwesig, I. Poupyrev, E. Mori, Gummi : a bendable computer, Proceedings of CHI'2004, ACM, 263—270, 2004.

[43] B. Shneiderman, *Technology-Mediated Social Participation : The Next 25 Years of HCI Challenges*, HCI International, 2011.

[44] G. Shoemaker, A. Tang, K. Booth, Shadow reaching : a new perspective on interaction for large displays, Proc. User Interface Software and Technology 2007 (UIST 2007), ACM Press, 2007.

[45] L. Suchman, *Plans and Situated Actions*, Cambridge University Press, 1987.

[46] M. Tauber, ETAG : Extended Task Action Grammar – A language for the Description of the User's Task Language, Proceedings INTERACT'90, 163–174, Elsevier, 1990.

[47] D. Vogel, P. Baudisch, Shift : a technique for Operating Pen-Based interfaces using Touch, in Proc. CHI 2007, ACM Pub., 2007.

[48] D. Widgor, C. Forlines, P. Baudisch, J. Bamwell, C. Shen, LucidTouch : a See-Through Mobile Device, in Proc. of UIST 2007, ACM Pub., 269-278, 2007.

[49] J. Yu, B. Benatallah, R. Saint-Paul, F. Casati, F.M. Daniel, M. Matera, A Framework for rapid Integration of Presentation Components, in proc. WWW'07 Proc. of the 16th International Conf. on World Wide Web, 923-932, 2007.