



HAL
open science

Évaluation des SI: besoins en méthodes et outils provenant de l'ergonomie et des IHM

Christophe Kolski, Houcine Ezzedine, Marie-Pierre Gervais, Káthia Marçal de
Oliveira, Ahmed Seffah

► **To cite this version:**

Christophe Kolski, Houcine Ezzedine, Marie-Pierre Gervais, Káthia Marçal de Oliveira, Ahmed Seffah. Évaluation des SI: besoins en méthodes et outils provenant de l'ergonomie et des IHM. XXXème Congrès INFORSID, INFormatique des ORganisations et Systèmes d'Information et de Décision, May 2012, Montpellier, France. pp.395-410. hal-03325780

HAL Id: hal-03325780

<https://uphf.hal.science/hal-03325780>

Submitted on 14 Feb 2024

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

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

Evaluation des SI

Besoins en méthodes et outils provenant de l'ergonomie et de l'IHM

Christophe Kolski*, **Houcine Ezzedine***, **Marie-Pierre Gervais ****,
Kathia Marcal de Oliveira*, **Ahmed Seffah*****

* Univ Lille Nord de France, F-59000 Lille
UVHC, LAMIH, F-59313 Valenciennes
CNRS, UMR 8201, F-59313 Valenciennes
{prénom, nom}@univ-valenciennes.fr

** Université Paris Ouest et Laboratoire d'Informatique de Paris 6
LIP6, F-75252 Paris cedex
Marie-Pierre.Gervais@lip6.fr

*** TechCICO, UMR STMR CNRS 8029
Université Technologique de Troyes, F-10010 Troyes cedex
ahmed.seffah@utt.fr

RÉSUMÉ. Cet article concerne l'évaluation des Systèmes d'information (SI), sous l'angle des besoins en processus adaptés, de même qu'en méthodes et outils provenant de l'ergonomie et de l'interaction homme-machine (IHM). Les besoins généraux sont d'abord mis en avant, de même que les besoins en évaluation dans les procédés logiciels. Les catégories de méthodes d'évaluation provenant de l'ergonomie logicielle et de l'IHM sont étudiées. Puis l'article se focalise sur les outils automatiques et semi-automatiques. Une discussion porte enfin sur l'étude des apports de tels méthodes et outils d'évaluation pour le domaine des SI.

ABSTRACT. This paper concerns IS evaluation, according to two points of view: needs in adapted software processes, needs in methods and tools from software Ergonomics and Human-Computer Interaction (HCI). General needs are first underlined, as well as needs concerning the place of evaluation in software processes. Several categories of evaluation methods from software Ergonomics and Human-Computer Interaction are studied. Then the paper is focused on needs concerning automated and semi-automated tools. Finally a discussion concerns the study of the contribution of such methods and tools for SI evaluation.

MOTS-CLÉS : Système d'information (SI), Evaluation, Méthode d'évaluation, Outil d'évaluation, Ergonomie logicielle, Interaction homme-machine.

KEYWORDS: information system (IS), Evaluation, Evaluation method, Evaluation tool, Software ergonomics, Human-Machine Interaction.

1. Introduction

De nombreuses études ont démontré qu'une mauvaise interface homme-machine peut provoquer le rejet de la part des utilisateurs voire de l'ensemble des parties prenantes, leur frustration, voire leur anxiété, face au système d'information qu'ils ont à utiliser (Bias and Mayhew, 2005). Inversement, une interface homme-machine facile à utiliser et à apprendre (on parle à ce sujet d'utilisabilité) amplifie les sensations positives de succès et de contrôle. Elle contribue à la réduction des coûts et délais de formation tout en réduisant les coûts de maintenance. Landauer (2011) et Mayhew (1999) dressent un panorama exhaustif des coûts et bénéfices de l'ingénierie des IHM et de leur utilisabilité.

Au-delà de l'utilisabilité, c'est le gouffre entre comment le système fonctionne et les expériences des usagers, qui est au cœur des travaux sur l'interaction homme-machine (IHM) au sens large. C'est pourquoi on préconise une démarche d'ingénierie centrée sur les expériences des usagers qui signifie que l'utilisateur doit être «au cœur» du processus itératif de conception du service. Celui-ci distingue trois grandes étapes. La première étape consiste, en plus de la spécification des besoins fonctionnels et organisationnels, à analyser les expériences et comportements des utilisateurs et du contexte dans lequel ils doivent utiliser le service. Cette première phase définit aussi les exigences qualité du SI selon l'utilisateur incluant l'utilisabilité, la satisfaction et l'accessibilité. Ensuite, les différents choix de conception sont pondérés au regard de leur impact sur le comportement de l'utilisateur lorsqu'il utilise le service. Une première version du service est conçue sous forme de maquette propice à des évaluations précoces. C'est la phase de conception dans laquelle on accorde une place importante à l'interface homme-machine. Finalement, la qualité du service est évaluée en plaçant l'utilisateur en situation d'utilisation réelle du produit. C'est la phase d'évaluation pendant laquelle sont conduits des tests qui permettent de vérifier que le SI répond aux exigences initiales de qualité selon l'utilisateur et supporte les expériences usagers.

Durant les dernières décennies, un large éventail de méthodes d'évaluation a été proposé ; voir par exemple la taxonomie de Ivory et Hearst (2001). En parallèle, un effort de standardisation a aussi contribué à l'émergence de modèles de qualité liés à l'interaction homme-machine. Par exemple les normes ISO 9126 et ISO 9141 proposent une définition de l'utilisabilité comme facteur de qualité du logiciel.

Récemment la communauté SI a pris conscience de l'importance de la dimension IHM et des méthodes d'ingénierie associées dans les systèmes d'information. L'indicateur le plus significatif de cette tendance est le nouveau journal Transactions on HCI lancé par l'AIS (Association of Information Systems), la plus importante société savante dans le domaine des SI. La question qui se pose alors est comment intégrer les travaux sur l'IHM et les approches actuelles d'ingénierie et de management des SI. Un point incontournable alors est l'évaluation du SI selon le point de vue de l'IHM. Cet article est une contribution à cet effort.

La partie 2 de l'article est consacrée aux SI et à leurs besoins en évaluation, de manière générale. Par la suite, les trois parties suivantes se focalisent successivement sur : les besoins en évaluation dans les procédés logiciels (partie 3), les besoins en méthodes d'évaluation centrées utilisateur (partie 4), les besoins en outils automatiques ou semi-automatiques pour l'évaluation des SI (partie 5). La partie 6 propose une discussion relative à l'étude des apports des méthodes et outils d'évaluation en IHM pour le domaine des SI. La dernière partie est consacrée à la conclusion et aux perspectives de recherche.

2. Des SI à leurs besoins en évaluation

Les SI sont de plus en plus complexes et interconnectés, tout en utilisant différentes technologies pour répondre à une large variété des besoins afin de soutenir les activités quotidiennes de différentes catégories d'utilisateurs. Cette section souligne, à partir de la définition d'un SI, l'importance de son évaluation et détaille l'évaluation qu'il est important de démarrer le plus précocement possible, dès les étapes d'analyse et conception.

De nombreuses définitions des SI ont été proposées dans la littérature. Reix définit un SI comme étant « un ensemble organisé de ressources : matériel, logiciel, personnel, données, procédures permettant d'acquérir, traiter, stocker, communiquer des informations (sous forme de données, textes, images, sons, etc.) dans les organisations » (Reix, 2002). C'est grâce à un SI que l'information est acquise, stockée, transmise et traitée entre les différents acteurs au sein de l'organisation. Il s'avère que les SI deviennent le principal véhicule de communication et d'intégration entre les différents acteurs au sein de l'organisation répondant à plusieurs intérêts.

Considérant l'importance grandissante accordée aux SI, de nombreuses entreprises procèdent à des grands investissements dans le développement et aussi dans l'optique d'utilisation d'un SI. Dans ce contexte, nous trouvons la large utilisation de systèmes pour la gestion intégrée, de type ERP (*Enterprise Resource Planning*), la gestion de la relation client (CRM - *Customer Relationship Management*), celle de la chaîne logistique (SCM - *Supply Chain Management*), celle des ressources humaines (HRM - *Human Resource Management*), entre autres. En conséquence, une gestion efficace et efficiente s'avère indispensable pour les entreprises qui doivent assurer la performance et la qualité des systèmes qu'ils utilisent. L'évaluation d'un SI est donc une nécessité pour les gestionnaires pour mieux justifier ces investissements. Smithson et Hirschheim (1998) résument cette question en indiquant que l'évaluation des SI est « le processus consistant à évaluer ou justifier la valeur des systèmes d'information, pour le but de prise de décisions organisationnelles à travers une sorte de discours de l'organisation ».

Selon Stockdale et Standing (2006) une évaluation effective doit répondre à cinq questions : (i) qu'est-ce qui est évalué, (ii) pourquoi procéder à l'évaluation, (iii) qui affecte l'évaluation et comment, (iv) quand réaliser l'évaluation et (v) comment

l'effectuer. Chacune de ses questions révèle des aspects importants à explorer. Par exemple, le premier aspect (« ce qui ») consiste à identifier quels objets étudier (les processus ou les livrables d'un système d'information) et sur quel critères (par ex. la satisfaction des utilisateurs, l'utilisabilité des systèmes, etc.). Le « qui évalue » souligne l'importance des utilisateurs et, notamment, la préoccupation de produire des systèmes les plus adaptables possible à leurs usages. Le « quand » et le « comment » évaluer » ravivent l'argument connu d'évaluer tôt dans le processus et en utilisant des paradigmes/méthodes appropriés. Ainsi, il est également important d'évaluer les processus logiciels utilisés autrement (aspect sur lequel se focalise la partie 3). Finalement, pour répondre au « pourquoi », on doit considérer des aspects politiques, économiques, culturels et sociaux de l'organisation (qui ne sont pas traités dans cet article).

3. Besoins en évaluation dans les procédés logiciels

Les éditeurs de logiciels visent continuellement à améliorer la qualité et la productivité du développement logiciel. Pour ce faire, ils s'appuient sur l'utilisation de procédés logiciels. Ceux-ci décrivent les bonnes pratiques que doit suivre une entreprise pour mener à bien les tâches de développement de logiciels (Humphrey, 1989). Il est important déjà ici de noter qu'à notre connaissance, la bonne pratique en termes d'évaluation exploitant des méthodes provenant de l'ergonomie et/ou de l'IHM est largement sous-exploitée dans la plupart des procédés logiciels actuels.

Différentes formes ont été utilisées pour représenter les procédés logiciels. Les modèles de cycle de vie se sont rapidement imposés (cascade, V, spirale, itératif, etc.) (Sommerville, 2010). Toutefois, ils restent à un niveau de description conceptuel et abstrait. Ainsi, ils sont essentiellement utilisés dans la définition de l'organisation, de la planification, ou encore de la détermination des coûts d'un projet. Des auteurs ont montré qu'une représentation plus détaillée que celle d'un modèle de cycle de vie était nécessaire pour décrire un procédé logiciel (Curtis *et al.*, 1992) ; en particulier, l'importance d'un enrichissement de modèles existants sous l'angle de l'IHM a été mise en avant depuis plusieurs années, voir par exemple Kolski *et al.* (2001) ou Lepreux *et al.* (2003). Un modèle de procédé logiciel représente un procédé logiciel au moyen d'un ensemble d'activités de procédé partiellement ordonnées, qui doivent être exécutées par les agents afin de produire et de maintenir un ensemble d'artefacts (Longchamp, 1993). L'intérêt d'un tel modèle est qu'il peut être analysé et exécuté. L'exécution d'un modèle de procédé consiste à améliorer la coordination entre les développeurs¹ d'un projet en assignant automatiquement les activités aux rôles appropriés, à router les artefacts d'une

¹ Dans cet article, le terme « Développeur » doit être pris au sens large, un des rôles qu'il nous paraît important de mettre désormais en avant est celui d'évaluateur (exploitant des méthodes de l'ergonomie et/ou de l'IHM).

activité à l'autre et à assurer le respect des échéances des activités. Une telle exécution est assurée par un PSEE (*Process-centered Software Engineering Environment*). Le rôle de ce dernier est également d'imposer que les développeurs respectent l'ordre d'exécution des activités et d'assurer qu'ils délivrent des artefacts corrects. Une manière de s'en assurer est de procéder à différentes évaluations, pouvant être réalisées de manière très précoce dans les procédés logiciels, pas seulement sur des maquettes ou prototypes, mais aussi sur des modèles du GL et de l'IHM (Bernonville *et al.*, 2010).

Ainsi les modèles de procédé et modèles d'artefacts, en tant que guides de développement, sont susceptibles d'améliorer la qualité de ce développement logiciel. Toutefois, la question se pose d'évaluer leurs apports respectifs : utilisés conjointement, permettent-ils une amélioration significative par rapport à une utilisation isolée de l'un ou de l'autre, voire à aucune utilisation de tels guides ? La littérature rapporte des expériences menées pour mesurer l'impact de l'utilisation de ces modèles en termes de temps et de qualité de conception (Cass *et al.*, 2005), (Almeida *et al.*, 2010). Les résultats obtenus sont cependant mitigés. En effet, selon le cas d'étude considéré, i.e., les activités menées par les agents en appliquant ces modèles, ils peuvent être diamétralement opposés. Il s'avère donc impossible d'obtenir des résultats généraux quant au bénéfice d'utiliser de tels modèles. L'évaluation de la qualité des développements logiciels fondés sur l'utilisation de ces modèles reste donc une question ouverte. Elle passe par l'identification de familles d'activités partageant des caractéristiques communes au sein desquelles les mesures seraient reproductibles. Il serait alors possible de déterminer des types d'activités pour lesquels l'utilisation conjointe d'un modèle de procédé et de modèles d'artefacts améliore le temps et la qualité du développement. Parmi ces types d'activité, comme précisé plus haut, l'évaluation est très souvent négligée, voire inexistante.

Par ailleurs, même lorsque les agents disposent d'un modèle de procédé leur servant de guide, il est reconnu qu'ils ne le respectent pas à 100% (Lanubile *et al.*, 2000). Ceci est dû au fait que les développeurs sont confrontés à des situations non capturées par le modèle de procédé, à des contraintes du projet tardives ou inattendues (pouvant prendre la forme d'insatisfaction utilisateur) ou bien qu'ils souhaitent accomplir leurs activités selon leurs expériences et intuitions. Un agent peut ainsi dévier du modèle de procédé. Le problème est alors pour le PSEE d'évaluer l'impact des déviations pour la suite du projet et d'identifier les actions appropriées pour éviter un échec.

Illustrons le problème sur l'exemple de modèle de procédé de la Figure 1, représenté par un diagramme d'activité UML (mais tout autre langage de modélisation de procédé pourrait être utilisé). L'exemple est composé de trois activités, A, B et C dont l'ordre d'exécution est défini par le flot de contrôle matérialisé par les flèches en gras alors que les autres flèches représentent le flot de données, i.e., les artefacts produits et consommés par les activités. Ainsi par exemple, l'activité C ne peut commencer si B n'est pas terminée et l'artefact A n'a pas été produit par l'activité A. Posons la déviation suivante : l'agent commence directement l'activité B, sans avoir fait A (qui peut correspondre à une activité d'évaluation précoce). Quels peuvent être les impacts d'une telle déviation pour la suite du procédé ? Examinons quelques options :

- Ignorer l'activité A. Cette option pose toutefois le problème de l'activité C qui consomme l'artefact A produit par A. Si A n'a pas lieu, l'artefact n'existe pas et C ne peut se dérouler. Un agent qui exécutera C commettra alors une nouvelle déviation au regard du modèle du procédé, dont il conviendra d'évaluer l'impact.

- Ignorer l'activité A en posant l'hypothèse qu'une autre activité assurera les actions que A devait faire, par exemple, produire l'artefact A. Si sur un exemple aussi simple que celui de la Figure 1, il est aisé d'identifier les dépendances, sur un modèle plus réaliste, cela s'avère rapidement fastidieux.

- Considérer que l'activité A sera exécutée ultérieurement lors du déroulement du procédé. Cela revient à considérer que A peut être différée mais toutefois est indispensable pour la bonne exécution du procédé. Par exemple, retarder une évaluation peut entraîner des activités inutiles, sous forme la conception ou la réalisation de parties de système interactif ne répondant pas ou insuffisamment aux besoins des utilisateurs.

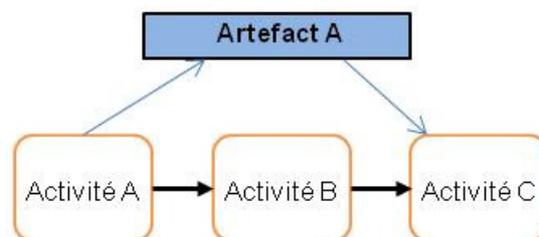


Figure 1. Un exemple de modèle de procédé logiciel

Actuellement, les PSEEs existants interdisent les déviations ou s'ils les tolèrent, c'est sous réserve que cette option était mentionnée dans le modèle de procédé (Cugola, 1998 ; Kabbaj, 2008). Estimer les impacts d'une déviation tels que ceux mentionnés ci-dessus afin de déterminer comment les traiter reste donc une question ouverte pour laquelle peu de propositions existent (Almeida *et al.*, 2011). En outre, comme précisé plus haut, un besoin se fait sentir dans les PSEE concernant

les évaluations centrées utilisateur. C'est pourquoi la partie suivante présente les méthodes disponibles à ce sujet, selon différentes catégories.

4. Besoins en méthodes d'évaluation centrées utilisateur

Dans les procédés logiciels, il s'avère que des évaluations peuvent être effectuées de manière très précoce dans les projets. Des évaluations peuvent être aussi effectuées progressivement sur des prototypes de plus en plus évolués. D'autres types d'évaluations peuvent aussi être effectués quand le SI est réalisé, proche de l'exploitation. Un besoin se fait donc sentir relativement aux types de méthodes utilisables dans les procédés logiciels. Pour répondre à ce besoin, une étude de classifications actuelles est nécessaire.

Classifier les méthodes d'évaluation centrées utilisateur, c'est-à-dire focalisées pour ce qui nous concerne sur l'utilisation des systèmes d'information (vus en tant que systèmes interactifs) n'est pas une tâche facile. En effet selon Bastien et Scapin (2001), la classification devrait permettre de décrire de manière exhaustive et adéquate les méthodes d'évaluation existantes afin d'en faciliter la sélection en fonction de critères tels que :

- les objectifs d'évaluation : s'agit-il d'un diagnostic, c'est-à-dire de détection des erreurs de conception en vue de fournir des alternatives de conception, ou d'évaluation permettant de déterminer jusqu'à quel point le système interactif est adapté pour les tâches pour lesquelles il a été conçu, ou encore d'une évaluation de conformité à des normes ?
- la source des données de l'évaluation : quelles sont les performances utilisateurs ? les caractéristiques de l'interface ? de l'interaction ? etc. ;
- le moment de l'évaluation : qui détermine l'état, la forme, la représentation du système interactif à évaluer. Par exemple on peut se trouver en présence d'une spécification, d'une maquette, d'un prototype, etc.

On peut facilement constater que plusieurs axes de classification des méthodes d'évaluation sont possibles. Notre objectif n'est pas de critiquer et d'analyser les classifications existantes en vue d'en proposer une nouvelle, mais d'illustrer les classifications parmi les plus connues dans la littérature afin de décrire les méthodes d'évaluation usuelles tout en les classant selon des approches. A ce sujet, plusieurs approches ont été proposées pour la classification des méthodes d'évaluation des systèmes interactifs : Senach (1990) distingue les méthodes d'évaluation suivant les données comportementales (méthodes empiriques) ou suivant les données sur l'interface (méthodes analytiques) ; Whitefield (1991), une des sources d'inspiration importantes de (Huart *et al.*, 2008) classe les méthodes d'évaluation selon la présence réelle ou représentée de l'utilisateur et du système ; Holyer (1993) introduit une classification selon la discipline (respectivement la psychologie cognitive, la psychologie sociale, les sciences sociales, l'ingénierie) qui sert de fondement aux méthodes prises en compte ; Balbo (1994) classe les méthodes selon plusieurs

critères éventuellement décomposés en sous-critères (connaissances requises, ressources matérielles, facteurs situationnels, moyens humains, résultats fournis par l'évaluation) ; Farenc (1997) sépare les approches d'évaluation selon trois éléments centraux de l'interaction homme-machine, que sont l'utilisateur, la tâche et le système. Bastien et Scapin (2001) proposent deux grandes catégories : les méthodes requérant la participation directe des utilisateurs et celles s'appliquant aux caractéristiques de l'interface, se distinguant (absence de l'interaction directe entre l'utilisateur et le système dans le cadre de l'évaluation).

Notons que, quelle que soit la classification, les méthodes les plus usitées sont recensées par la norme ISO/TR 16982:2002 « Ergonomie de l'interaction homme-système - Méthodes d'utilisabilité pour la conception centrée sur l'opérateur humain ». Elles sont le plus souvent centrées sur les critères d'utilité et d'utilisabilité (Nielsen, 2003).

Dans cet article, nous nous basons sur la classification de Grislin et Kolski, (1996), qui classe les méthodes d'évaluation en trois grandes approches (Fig. 5) :

- les approches centrées sur les avis et/ou les activités des utilisateurs,
- les approches qualifiées d'expertes centrées sur le jugement d'experts en communication homme-machine ou sur l'utilisation de grilles d'évaluation ou de questionnaires listant les qualités d'une bonne IHM,
- les approches qualifiées d'analytiques, centrées sur une modélisation de l'IHM et/ou de l'interaction homme-machine. Celles-ci consistent le plus souvent à effectuer l'évaluation à l'aide de métriques objectives à partir d'un modèle descriptif des tâches humaines, ou à partir d'une description des pages-écrans.

Toutefois comme le soulignent Abed (2001) ou encore Bastien et Scapin (2001), il paraîtrait irréaliste de considérer ces regroupements de manière indépendante. En effet, des relations transversales entre méthodes sont souvent très utiles. Par exemple, lors de l'analyse de situations de références avec des utilisateurs, il est souvent utile de procéder à une modélisation des tâches dans un but de spécification ; mais il est aussi possible de réutiliser cette modélisation lorsque le système interactif est réalisé, afin de la confronter à une modélisation des activités réelles des utilisateurs.

Notre objectif dans cette partie n'était pas de faire un recensement exhaustif de toutes les méthodes d'évaluation existantes dans la mesure où on retrouve plusieurs documents qui traitent ce sujet, mais plutôt de présenter les catégories de méthodes d'évaluation les plus représentatives. Un besoin important réside dans l'exploitation d'outils automatiques ou semi automatiques venant en support des méthodes, particulièrement aux approches empiriques de diagnostic d'usage (Figure 2). Ils font l'objet de la partie suivante.

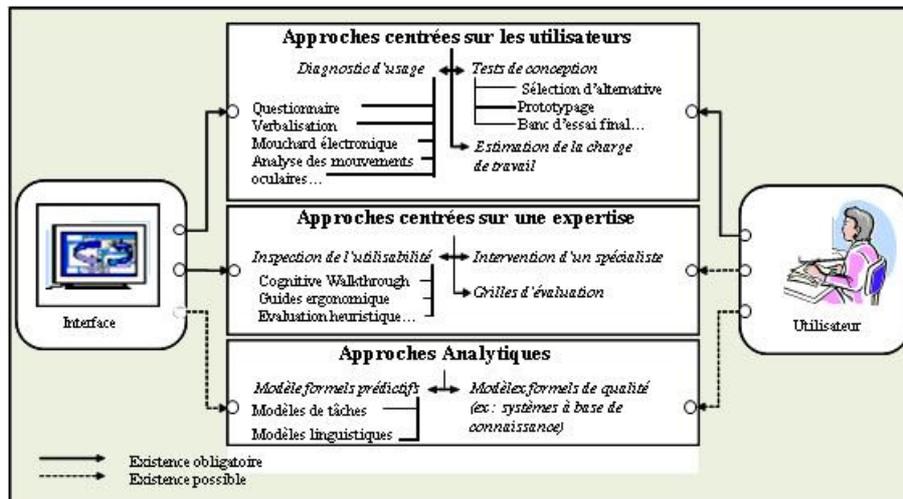


Figure 2 : Extrait de la classification des méthodes et techniques d'évaluation adaptant (Grislin et Kolski, 1996 ; Trabelsi, 2006)

5. Besoins en outils automatiques ou semi-automatiques pour l'évaluation des SI

L'automatisation de l'évaluation présente plusieurs avantages (Ivory et Hearst, 2001), correspondant à autant de besoins dans les procédés logiciels :

- réduire le coût de l'évaluation : les méthodes qui automatisent la capture, l'analyse, ou la critique (proposition de commentaires et d'alternatives) peuvent réduire le temps de l'évaluation et par conséquent le coût. Par exemple, les outils logiciels qui enregistrent automatiquement les événements (utilisateur/système) pendant l'évaluation éliminent le besoin de notation manuelle, qui nécessite un temps considérable ;

- faciliter la détection des erreurs : dans certains cas dans la mesure où l'interface peut être spécifiée selon des modèles de tâche (par exemple avec CTT (Paterno *et al.*, 1997), il devient possible de détecter automatiquement des incohérences et des erreurs, au niveau de l'interface à évaluer, par rapport à cette spécification ;

- réduire le besoin de recours à des experts en évaluation : automatiser quelques aspects d'évaluation, tels que les activités d'analyse ou de critique, apporte en principe une aide première aux concepteurs qui n'ont pas d'expérience approfondie en évaluation ;

- augmenter le recouvrement des aspects évalués : en raison du manque de temps, du coût élevé, et des contraintes en termes de ressources disponibles, il n'est pas toujours possible d'évaluer indépendamment chaque aspect de l'interface. Ainsi,

avec le recours à des outils logiciels d'évaluation automatique, il peut devenir possible de couvrir la totalité ou quasi-totalité des aspects de l'interface à évaluer.

Il est donc clair que les outils automatiques ou semi-automatiques peuvent être d'un apport intéressant pour l'évaluation des SI. Il s'agit donc d'étudier les différents types d'outils qui ont pu être proposés.

L'automatisation de l'évaluation des systèmes interactifs a débuté au début des années 80 avec le système Playback développé dans le service dédié aux facteurs humains des laboratoires IBM (Neal, 1983). Ce système était simple et permettait la capture des actions clavier au moyen d'un dispositif physique situé entre le clavier et la machine. En mode enregistrement, le dispositif était capable de détecter, transmettre, dater et sauvegarder sur disque toutes les actions de l'utilisateur sur le clavier. Le fichier contenant les sauvegardes alimentait un analyseur statistique pour, par exemple, détecter le nombre de sollicitations de la touche « aide » ou « détruire ».

Depuis, divers outils logiciels d'aide à l'évaluation ont été proposés. Selon Bastien et Scapin (2002) et Ivory et Hearst (2001), il existe trois types d'outils d'aide à l'évaluation :

- les outils constituant une version logicielle de documents papiers : par exemple, dans la méthode *Cognitive Walkthrough* (ou une de ses variantes, cf. Mahatody *et al.*, 2010) la tâche de l'évaluateur consiste à remplir des formulaires sur les actions des utilisateurs ; cette tâche devient souvent lourde avec des saisies répétitives d'informations souvent identiques. C'est essentiellement pour réduire cette charge de travail que des versions logicielles de la méthode ont été développées (Rieman *et al.*, 1991 ; Mahatody *et al.*, 2009). Elle permet d'automatiser de nombreuses saisies d'informations.

- Les outils d'accompagnement de l'évaluation : ceux-ci aident l'évaluateur à structurer et organiser l'évaluation. On retrouve dans cette catégorie (a) des outils d'aide au choix de la ou des méthodes d'évaluation à utiliser, ainsi que (b) des outils d'organisation de l'évaluation. Par exemple l'outil FACE (Fast Audit based on Cognitive Ergonomics) (Hulzebosch et Jameson, 1996) permet à l'évaluateur de cerner la situation de l'évaluation, de choisir la technique d'évaluation qui sera appliquée et de présenter les résultats.

- Les outils d'évaluation automatique proprement dits : ces outils permettent l'automatisation de l'évaluation selon trois approches : (a) la capture automatique, certains outils enregistrant différents types d'informations audiovisuelles, tels que la frappe au clavier ou la manipulation de la souris, (b) l'analyse automatique des données capturées dans un but le plus souvent de détection de problèmes d'utilisabilité ou d'accessibilité, (c) la critique automatique permet de localiser certains problèmes rencontrés et de fournir des propositions d'amélioration et des recommandations.

Parmi les outils automatiques pour l'aide à l'évaluation proposés depuis la fin des années 80, on peut citer sans souci d'exhaustivité : SYNOP (Kolski, 1989 ;

Kolski et Millot, 1991), KRI/AG (Lowgren et Nordqvist, 1992), CHIMES (Jiang *et al.*, 1992), ÉMA (Balbo, 1994), ERGOVAL (Farenc, 1997), EDEM (Hilbert et Redmiles, 1998), KALDI (Al-Qaimari et McRostie, 1999), Sherlock (Grammenos *et al.*, 2000), GUIDE (Henninger, 2000). Les outils les plus récents sont plutôt dédiés à l'évaluation automatique des sites Web, tels WebRemUSINE (Paganelli et Paternò, 2002), KWARESMI (Beirekdar, 2004), DESTINE (Mariage, 2005) ou WebTango (Cf. <http://webtango.berkeley.edu>).

De tels outils offrent des fonctionnalités de base (approches à base de connaissance, analyseurs de code) nous paraissant exploitables et adaptables pour l'évaluation de la partie interactive des SI, au même titre que certaines des catégories de méthodes présentées en partie 3. Ce point fait l'objet d'une discussion ci-dessous.

6. Discussion : étude des apports des méthodes et outils d'évaluation en IHM pour le domaine des SI

L'ergonomie logicielle et l'interaction Homme-Machine offrent toute une palette de méthodes d'évaluation, de même qu'un ensemble d'outils aussi bien automatiques que semi-automatiques. Cependant la majorité de ceux-ci sont le plus souvent proposés dans un but générique, afin de pouvoir être exploités dans différents domaines d'application.

Il est donc important de les reconsidérer complètement sous un nouvel angle de vue, celui des SI et de leurs spécificités. Afin d'aller vers une démarche réconciliant IHM et SI et dans un but d'intégration des méthodes d'évaluation des IHM dans le cycle de vie global du SI, les éléments suivants méritent un effort de recherche.

Premièrement, il serait important d'élaborer un modèle de qualité unifié des SI qui intègre l'utilisabilité et les facteurs classiques des SI comme la fiabilité, la sécurité et la durabilité ou encore l'évolutivité des SI. Le problème n'est plus de savoir combien coûte la mise en oeuvre d'un système d'information, mais plutôt combien pourrait coûter à l'entreprise un système inacceptable pour les usagers. Par exemple, il n'est plus suffisant d'assurer uniquement le fonctionnement sûr (au sens de la sûreté de fonctionnement) et sécuritaire d'un système d'information ; en effet il est important de se rendre compte combien pourrait coûter à l'entreprise un système qui présente une mauvaise IHM, et d'agir en conséquence à l'aide de procédés logiciels adaptés, intégrant des méthodes et outils d'évaluation.

Deuxièmement, l'application des méthodes d'évaluation en IHM exige une expertise dont l'absence peut compromettre l'efficacité des méthodes. Un effort de formation est requis par l'introduction de cours à ce sujet dans les cursus de formation de concepteurs et développeurs de SI. Du point de vue industriel, la place d'un expert en IHM que ce soit dans les équipes de développement des SI, d'assurance qualité, d'administration et de gouvernance des SI est aussi à clarifier. Mayhew (1999) et Seffah et Metzker (2004) proposent une démarche qui vise à

faciliter l'adoption et l'institutionnalisation de la fonction IHM/utilisabilité dans une organisation.

Troisièmement, du point de vue de beaucoup d'acteurs dans les projets (aussi bien du côté du maître d'ouvrage que de celui du maître d'œuvre), le système d'information d'une entreprise s'assimile encore parfois... au système informatique. Or, un système d'information inclut le système informatique mais ne s'y réduit pas. Même si les travaux de la communauté IHM ne traitent que la partie interface du système informatique et de sa partie interaction, les méthodes proposées sont applicables au système d'information. Par exemple, on peut penser à l'usage des méthodes d'évaluation de tous formulaires papiers qui sont une partie intégrante du SI.

7. Conclusion

Cet article a concerné l'évaluation des SI, domaine encore peu exploré, aussi bien au niveau national qu'international.

Après une première section traitant des SI et de leurs besoins généraux en évaluation, l'article s'est focalisé sur les besoins en évaluation dans les procédés logiciels. Il s'avère que l'évaluation est une activité encore peu présente dans les procédés logiciels, de même que dans les environnements de Génie Logiciel centrés procédés.

Pour couvrir les activités d'évaluation, l'article a ensuite passé en revue les catégories de méthodes d'évaluation centrées humain, provenant des domaines de l'ergonomie et de l'interaction homme-machine. L'exploitation de telles méthodes correspond à un besoin important pour l'évaluation des SI.

Parmi les différentes catégories de méthodes provenant des domaines de l'ergonomie et de l'interaction homme-machine, les outils automatiques et semi-automatiques ont été passés en revue. L'exploitation de telles méthodes offrent un potentiel important pour faciliter l'évaluation des SI.

La discussion a porté sur une étude des apports des méthodes et outils d'évaluation en IHM pour le domaine des SI et a mis en évidence différentes perspectives.

Cet article peut être considéré comme une pierre apportée au domaine de l'évaluation des SI, d'autres perspectives de recherche pouvant être envisagées :

- caractériser chaque méthode et outil relativement à un ensemble de spécificités (à déterminer finement) ayant des incidences, aussi bien positives que négatives, sur l'évaluation des SI ; il s'agit de porter un intérêt particulier aux méthodes et outils contribuant à l'évaluation précoce, voire même prédictive, du système d'information.

- porter une attention particulière à l'évaluation des systèmes d'information utilisés dans les domaines considérés comme critiques et/ou complexes (par exemple

les systèmes d'information utilisés dans les domaines de la santé, dans les transports (cf. Kolski *et al.*, 2011), de la production d'énergie, dans le domaine militaire, ...) ;

– étudier sur un ensemble de SI (benchmark à mettre en place) l'apport de différentes combinaisons de méthodes et outils d'évaluation, afin d'en comparer l'efficacité et le coût.

Remerciements

Les auteurs remercient l'association Inforsid pour le soutien apporté au groupe de travail sur l'évaluation des systèmes d'information, dont ils sont membres, de même que les relecteurs anonymes de la conférence pour leurs remarques constructive. Les deux premiers auteurs remercient également la région Nord-Pas de Calais, le FEDER (Communauté Européenne) et le CISIT.

8. Bibliographie

- Abed M., *Méthodes et modèles formels et semi-formels pour la conception et l'évaluation des systèmes homme-machine*. Habilitation à diriger des recherches, Université de Valenciennes et du Hainaut-Cambrésis, mai 2001.
- Almeida M.A., Blanc X., Bendraou R., « Deviation Management during Process Execution », *Proceedings of the ACM Internat' Conf' on Automated Software Engineering*, 2011, p. 528-531.
- Almeida M.A., Mougnot A., Bendraou R., Robin J., Blanc X., « Artifact or Process Guidance, an Empirical Study », *Proceedings of the 13th Internat' Conf' MODELS 2010*, Lecture Notes in Computer Science n° 6395, 2010, p. 318-330.
- Balbo S., *Evaluation ergonomique des interfaces utilisateur : un pas vers l'automatisation*. Thèse de doctorat, université Joseph Fourier, Grenoble I, septembre 1994.
- Bastien J.M.C., Scapin D.L., Les méthodes ergonomiques : de l'analyse à la conception et à l'évaluation. In : *Ergonomie et Informatique Avancée, Ergo-IA'2002*, A. Drouin, J. Robert (Eds), I.D.L.S., Biarritz, 8-10 octobre 2002.
- Bastien J.M.C., Scapin L., Evaluation des systèmes d'information et critères ergonomiques. In Kolski C. (Ed.), *Environnement évolués et évaluation de l'IHM. Interaction Homme Machine pour les SI*, Volume 2, p. 53-80. Paris : Hermès, 2001.
- Bernonville S., Kolski C., Leroy N., Beuscart-Zéphir M., Integrating the SE and HCI models in the human factors engineering cycle for re-engineering Computerized Physician Order Entry systems for medications: basic principles illustrated by a case study. *International Journal of Medical Informatics*, 79 (4), pp. 35-42, 2010.
- Bias R.G., Mayhew D.J. (Eds.), *Cost-Justifying Usability, Second Edition: An Update for the Internet Age*, Morgan Kaufmann, 2005.

- Cass A.G., Osterweil L., « Process support to help novices design software faster and better », *Proceedings of the ACM Internat' Conf' on Automated Software Engineering*, 2005, p. 295-299.
- Cugola G., « Tolerating deviations in process support systems via flexible enactment of process models », *Communications of the ACM*, vol. 35, n° 9, 1992, p. 75-90.
- Curtis B., Kellner M.I., Over J., « Process modeling », *IEEE Transactions On Software Engineering*, vol. 24, n° 11, 1998, p. 982-1001.
- Farenc C., *ERGOVAL: une méthode de structuration des règles ergonomiques permettant l'évaluation automatique d'interfaces graphiques*. Thèse de Doctorat de l'Université Toulouse 1, janvier 1997.
- Grislin M., Kolski C., « Evaluation des interfaces homme-machine lors du développement de système interactif ». *Technique et Science Informatiques (TSI)*, 3, p. 265-296.
- Holyer A., *Methods for evaluating User Interfaces. School of Cognitive and Computing Sciences*. University of Sussex, Brighton, 1993.
- Huart J., Kolski C., Bastien C., L'évaluation de documents multimédias, état de l'art. In Merviel S. (Ed.), *Objectiver l'humain ? Volume 1, Qualification, quantification*, Hermes, Paris, p. 211-250, 2008.
- Hulzebosch R., Jameson A. FACE: A Rapid Method for Evaluation of User Interfaces. In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & I. L. McClelland (Eds.), *Usability evaluation in industry*, p. 195-204, London, 1996.
- Humphrey W.S., *Managing the Software Process*, Boston, Addison-Wesley, 1989.
- Ivory M., Hearst M., « The State of the Art in Automated Usability Evaluation of User Interfaces. *ACM Computing Surveys*, 33 (4), 2001, p. 173-197.
- Jiang J., Murphy E.D., Bailin S.C., Truszkowski W.F., Szczur M., Prototyping a Knowledge based compliance checker for User-Interface. Evaluation on Motif development environments. *Second Annual International Motif Users Meeting*, Bethesda, MD: Open Systems, p. 258-268, 1992.
- Kabbaj M., Lbath R., Coulette B., « A deviation management system for handling software process enactment evolution », *Proceedings of the Internat' Conf' on Software Process*, Lecture Notes in Computer Science n° 5007, 2008, p. 186-197.
- Kolski C., *Contribution à l'ergonomie de conception des interfaces graphiques homme-machine dans les procédés industriels : application au système expert SYNOP*. Thèse de Doctorat, Université de Valenciennes et du Hainaut-Cambrésis, Janvier 1989.
- Kolski C., Ezzedine H., Abed M. (2001). Développement du logiciel : des cycles classiques aux cycles enrichis sous l'angle des IHM. In Kolski C. (Ed.), *Analyse et Conception de l'IHM. Interaction Homme-machine pour les SI*, vol. 1, Hermès, Paris, pp. 23-49, ISBN 2-7462-0239-5, 2001.
- Kolski C., Millot P., « A rule-based approach to the ergonomic "static" evaluation of man-machine graphic interface in industrial processes ». *International Journal of Man-Machine Studies*, 35, p. 657-674, 1991.

- Kolski C., Uster G., Robert J-M., Oliveira K., David B., Interaction in mobility: the evaluation of interactive systems used by travellers in transportation contexts. In J.A. Jacko (Ed.), *Human-Computer Interaction: Towards Mobile and Intelligent Interaction Environment, 14th International Conference, HCI International 2011, Orlando, FL, USA, July 9-14, 2011, Proceedings, Part III*, Lecture Notes in Computer Science 6763, Springer, p. 301-310, 2011.
- Lanubile F., Visaggio G., Evaluating defect detection techniques for software requirements inspections, ISERN-00-08, International Software Engineering Research Network, 2000.
- Lepreux S., Abed M., Kolski C., « A human-centred methodology applied to decision support system design and evaluation in a railway network context ». *Cognition Technology and Work*, 5, p. 248-271, 2003.
- Lonchamp J., « A structured conceptual and terminological framework for software process engineering », *Proceedings of the International Conference on Software Process*, 1993, p. 41-53.
- Lowgren J., Nordqvist T., Knowledge-based evaluation as design support for graphical user interfaces. *Proceeding of ACM conference on Human Factors in Computing Systems, CHI'92*, Bayersfeld, (Eds.), Monterey, California, p.181-188. May 3-7, 1992.
- Mahatody T., Kolski C., Sagar M., CWE: Assistance Environment for the Evaluation Operating a Set of Variations of the Cognitive Walkthrough Ergonomic Inspection Method. In D. Harris (Ed.), *Engineering Psychology and Cognitive Ergonomics, 8th International Conference, EPCE 2009, Held as Part of HCI International 2009* (San Diego, CA, USA, July 19-24, 2009), Proceedings, LNAI 5639, Springer-Verlag, p. 52-61.
- Mahatody T., Sagar M., Kolski C., State of the Art on the Cognitive Walkthrough method, its variants and evolutions. *International Journal of Human-Computer Interaction*, 26 (8), p. 741-785, 2010.
- Mayhew D.J., *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*, Morgan Kaufman Publishers, 1999.
- Neal A.S, Simons R.M., Playback: a method for evaluating the usability of software and its documentation. *Proceedings of the CHI'83 Conference on Computer-Human Interaction*, ACM New York, p. 78-82, December 1983
- Nielsen J., *Usability Engineering*. Academic Press, Boston, 1993.
- Paterno F., Mancini C., Meniconi S., ConcurTaskTrees: A Diagrammatic Notation for Specifying Task Models, *Proceedings Interact'97*, p. 362-369, Sydney, Chapman&Hall, 1997.
- Reiman J., Davies S., Hair C., Esemplare M., Polson E., Lewis C., An automated cognitive walkthrough. In Robertson, S.E, Olsen, G.M., and Olsen, J.S. (Eds.), *Human Factors in Computing Systems, CHI 91 Conference Proceedings*, 1991, ACM Press, New York.
- Reix R., *Systèmes d'information et management des organisations*, 4^e éd., Vuibert, 2002.
- Seffah A., Metzker E. « Usability and Software Engineering Prevalent Myths and Obstacles », *Communications of the ACM*, 47 (12), p. 71-76, 2004

- Senach S., *Evaluation ergonomique des IHM : Une revue de la littérature*. Rapport INRIA, n°1180, mars, 1990.
- Smithson S., Hirschheim, R. « Analysing information system evaluation: another look at an old problem ». *European Journal of Information Systems*, 7(3), p. 158-174, 1998.
- Sommerville I., *Software engineering*, 9th ed., Addison-Wesley, 2010.
- Stockdale R., Standing C. « An interpretative approach to evaluating information systems: a content, context, process framework ». *European Journal of Operational Research*, 173, p. 1090-1102, 2006.
- Trabelsi A., *Contribution à l'évaluation des systèmes interactifs orientés agents, application à un poste de supervision de transport urbain*. Thèse de Doctorat, Université de Valenciennes et du Hainaut-Cambrésis, Septembre 2006.
- Whitefield A., Wilson F., Dowell J., « A framework for human factors evaluation ». *Behaviour & Information Technology*, Vol 10, n°1, p. 65-79, 1991.