

Systèmes Interactifs en Co-évolution

Réflexions sur les apports de la Théorie de l'Activité au support des Pratiques Collectives Distribuées

Co-evolution inside Interactive Systems
Reflections about results coming from the Activity Theory for supporting Collective and Distributed Practices

Grégory BOURGUIN (1), Alain DERYCKE (2)

- (1) Laboratoire d'Informatique du Littoral (LIL),
Université du Littoral Côte d'Opale, France
- (2) Laboratoire Trigone,
Université des Sciences et Technologies de Lille, France
bourguin@lil.univ-littoral.fr, alain.derycke@univ-lille1.fr

Résumé. Les méthodes traditionnelles de conception des systèmes interactifs, qu'elles soient basées sur les tâches ou orientées objet, révèlent de plus en plus des lacunes compte tenu des évolutions actuelles. Ces évolutions sont à la fois extérieures aux systèmes développés, par exemple des évolutions au sein des organisations, et intrinsèques à ces systèmes, par exemple les évolutions technologiques. Deux des plus grands reproches que l'on peut adresser à ces méthodes sont leur rigidité et leur linéarité. D'une part, dès lors que les systèmes développés ont été implantés sur site, la seule évolution que l'on peut leur faire subir se fait au travers de la maintenance. D'autre part, pour aboutir à ces systèmes, la conception se déroule suivant des étapes qui segmentent le temps, les acteurs, et les travaux à réaliser. De manière à pallier ces problèmes, nous mettons en œuvre des résultats issus de la Théorie de l'Activité qui nous ont permis d'aboutir aux concepts de Système Interactif Evolutif et de Co-évolution. Ces travaux nous ont amenés à repenser le concept de tâche dans son rapport à l'activité qui en découle et à développer une infrastructure informatique en adéquation avec les mécanismes liés à l'évolution des activités qu'elle tente de supporter.

Mots-clés. Co-évolution, système interactif, Théorie de l'Activité, tâche, réflexion.

Abstract. Traditional task based or object oriented design methods for interactive systems reveal some lacks regarding nowadays evolutions. Evolution can appear outside the developed system like inside the organisations. It can also be intrinsically linked to the system like a technological evolution. The two main criticisms that can be done to these traditional approaches are their rigidity and their linearity. On one hand, once the system has been developed and implanted, it can only be evolved through maintenance. On the other hand, the employed design methods usually

introduce diverse steps splitting the time, the actors and their work. Trying to palliate to these problems, we use some results coming from the Activity Theory that helped us in understanding what should be an Evolving Interactive System and defining the Co-evolution principle. This work leads us to rethink the place of the task concept in its relation with the resulting activity. We then have developed a computer infrastructure responding to the evolution mechanisms that are linked to the human activities we want to support.

Keywords. Co-evolution, interactive system, Activity Theory, task, reflection.

1 Introduction

Depuis de nombreuses années nous sommes engagés dans la conception d'interfaces pour des systèmes interactifs complexes destinés à supporter, au travers de l'Internet et du Web, des activités humaines menées collectivement et distribuées du point de vue géographique, des activités encore appelées des Pratiques Collectives Distribuées ou PCD. Notre cadre de recherche est donc situé à l'intersection des domaines des IHM et des PCD, avec deux champs d'applications visés : les systèmes interactifs pour l'apprentissage flexible et à distance, que cet apprentissage soit individuel ou collaboratif comme dans le Computer Supported Collaborative Learning (Derycke, Kaye, 1993), (Bourguin, Derycke, 2000) dans le cadre du laboratoire TRIGONE ; et plus récemment les systèmes interactifs destinés à supporter les processus de conception collaborative de logiciels au sein des équipes de développement, un cadre particulier d'application du Travail Coopératif Assisté par Ordinateur (ou TCAO) au sein du LIL.

Il nous est apparu, dès nos premiers développements entre 1992 et 1998, que nous devons concevoir des infrastructures technologiques ayant une malléabilité fonctionnelle, c'est-à-dire susceptibles d'être adaptées localement aux usages envisagés et aux différences interindividuelles, ainsi qu'une malléabilité organisationnelle pour s'adapter aux différents contextes organisationnels rencontrés. Nous fondions cette demande sur les premières critiques faites aux premiers systèmes de TCAO : par exemple celles de Johnson-Lenz (1991) qui incitaient les concepteurs de systèmes à laisser du « jeu » dans les mécanismes mis en œuvre dans les collecticiels. Nos premières conceptions de tels environnements essayaient de fournir des solutions pour ces deux formes de malléabilité, notamment à partir de l'utilisation de modèles conceptuels centrés sur l'activité (Derycke, 1998) et isolant les rôles des acteurs (droits & devoirs) par rapport aux outils et aux données, ou encore, par la possibilité de modifier les scripts régissant certains comportements du système. Mais cette souplesse n'était accessible qu'aux concepteurs, et seulement lors de la génération d'une instance spécifique de notre plate-forme, en réponse à un besoin identifié et bien spécifié. Lors de nos mises en œuvre expérimentales, il est apparu que les divers acteurs souhaitaient pouvoir accommoder le système existant (et opérationnel) pour tenir compte des changements intervenus tant dans le contexte organisationnel de leurs activités, que dans la définition de l'activité elle-même, pour y refléter les adaptations aux évolutions des compétences des acteurs.

Ceci nous a amenés à revoir en profondeur la conception de tels Systèmes pour qu'ils soient Interactifs et Evolutifs (des SIE dans la suite du texte), et à chercher de nouveaux fondements théoriques sur lesquels établir une nouvelle approche de la conception de ces SIE, en mettant à profit l'expérience trans-disciplinaire acquise ces dernières années dans nos relations, en tant que concepteurs, avec les sciences sociales et du comportement humain. C'est cette nouvelle approche que nous allons

présenter maintenant en nous focalisant essentiellement sur les fondements tels que nous les avons dérivés des courants théoriques issus des sciences sociales et de la psychologie, renouvelant ainsi ce qui a toujours nourri la science des IHM depuis son origine. Les aspects plus logiciels des propositions qui vont suivre ne seront ici qu'évoqués par souci de clarté. Nous avons en effet souhaité mettre en avant les dimensions plus conceptuelles et modélisantes de notre travail, et les évolutions historiques de notre conception elle-même, au cours des différentes mises en œuvre ou mises à l'épreuve. Cette approche que l'on pourrait qualifier de « *Theory-Based* » est en effet assez complexe et nécessite une présentation assez détaillée du cadre théorique, la Théorie de l'Activité dans notre cas, pour que le lecteur puisse comprendre comment ce cadre théorique et les modèles dérivés sont en quelque sorte réifiés au sein des systèmes.

Notre article présente d'abord nos motivations et les évolutions au cœur même de la science des IHM qui justifient une nouvelle approche pour la conception des SIE. Parmi les apports des sciences humaines, nous allons nous focaliser sur un cadre théorique, celui de la Théorie de l'Activité, qui s'est révélé très intéressant ces dernières années pour la communauté scientifique du TCAO. A partir de ce cadre et d'une première tentative de capturer les concepts les plus intéressants pour en faire des principes directeurs lors de la conception de nouvelles infrastructures destinées aux PCD, nous allons présenter deux temps de notre évolution illustrés chacun par un projet particulier d'infrastructure informatique : tout d'abord le projet DARE, puis le projet CoolDA qui représente une seconde itération suite aux enseignements tirés de la première conception et implémentation. Il est évident que cette nouvelle approche, ambitieuse et complexe, demande encore beaucoup d'efforts de réalisation avant de pouvoir être validée dans des situations réelles. Cependant, nous allons pouvoir d'ores et déjà tirer les premiers enseignements quant à l'intérêt de telles infrastructures évolutives fondées sur un corps de théories non techniques. Ces résultats peuvent alimenter les réflexions des concepteurs de systèmes interactifs dans une perspective plus anthropotechnologique (Rabardel, 1995).

2 Une nouvelle conception des IHM prenant en compte les phénomènes de co-évolution

2.1 Une limite des approches classiques des IHM dans la conception des systèmes d'information

Il apparaît que les développements récents des technologies de l'informatique et des réseaux de communication, en particulier avec le phénomène Internet, posent de nouveaux problèmes et enjeux à la communauté IHM. Une partie de ces problèmes réside dans la dimension de plus en plus sociale des usages, et de l'impact des systèmes d'information interactifs sortant largement de la sphère du professionnel. Les attentes sont fortes :

- tant pour le processus de conception des IHM lui-même où, malgré des progrès dans la technologie et dans des processus de conception informés par les apports de l'ergonomie cognitive, il subsiste de nombreux problèmes d'utilisabilité et d'usage ;
- qu'au niveau de secteurs d'application, avec une dynamique des besoins conduisant jusqu'à considérer les organisations humaines au sein desquelles les systèmes interactifs sont déployés, comme des organisations virtuelles ou apprenantes (Senge, 1990) et donc en perpétuelle évolution dans un monde incertain.

Dans (Bourguin, 2000) et (Bourguin, Derycke, Tarby, 2001), nous avons déjà affirmé que cela appelait une nouvelle conception des IHM dont le concept même était en quelque sorte remis en cause. Nous avons en particulier critiqué les approches traditionnelles des processus de conception des systèmes interactifs, fondées essentiellement sur une approche rationaliste et positiviste, qui induit des ruptures dans les processus : rupture entre processus de conception et processus d'appropriation collective (séparation concepteurs/usagers) et rupture, ou difficulté de jeter des ponts, entre les disciplines scientifiques mobilisées, notamment entre l'informatique et les sciences sociales (Bowker, Star, Turner, 1997).

De même nous avons montré que des démarches de conception¹ qui visent à prendre en compte ces dynamiques du changement des besoins des acteurs et des contextes techniques et organisationnels, et qui sont fondées sur la prise en compte des exceptions au fonctionnement normal, ne peuvent pas résoudre le problème sur le fond. En effet, la nature même de ces exceptions est imprédictible et le changement est la norme (Vicente, 2000). Cependant, dans le domaine des IHM et des stratégies de traitement des exceptions, le travail de Wild et Macredie (2000) avait retenu notre attention pour sa focalisation sur le traitement de la tâche qui est un concept central dans la conception des IHM. Wild et Macredie proposent en effet d'étendre certaines méthodes d'analyse de tâches afin de les rendre plus souples quant à l'introduction des changements. Ils introduisent le concept d'exception, quelque chose d'imprévu dans le modèle normatif de la tâche, qui constitue une autre source d'information sur les évolutions nécessaires de la tâche et sur l'altération des besoins du système. Nous montrerons dans la suite de l'article que notre approche est plus radicale en introduisant un rapport dialectique entre la tâche (son modèle en fait) qui sert de support à l'interaction et l'activité humaine singulière, située, socialement encadrée dans son contexte.

Les incitations à des démarches plus anthropocentriques

Notre critique et notre analyse rejoint celles de chercheurs qui ont, dès le milieu des années 80, proposé d'autres voies pour la conception des systèmes informatiques en prenant leurs valeurs et fondements dans d'autres disciplines : pour Winograd et Flores (1986) il s'agit de la philosophie et de la linguistique pragmatique. Pour Suchman (1987) il s'agit des sciences sociales avec les dérivées de la sociologie comme l'ethnographie et les théories de l'action située. Enfin plus récemment, Hollan et al (2000) montrent que la Cognition Distribuée peut elle aussi fournir un cadre théorique et expérimental à la re-fondation des IHM, notamment pour l'informatique ubiquitaire, et un cadre intégré pour la recherche dans ce domaine.

Dans le domaine du TCAO, les sciences sociales, notamment des branches de la sociologie, ont eu un impact important sur la communauté de recherche internationale. Mais comme le notent Hugues et al (1995), pour le cas de l'ethnographie, les résultats de l'analyse de besoins apparaissent, pour les technologues, comme insuffisamment génériques et souvent vagues, et se traduisent le plus souvent en contraintes pour la conception du système plutôt qu'en spécifications fonctionnelles (Sommerville et al., 94). Le cas de l'ethnométhodologie est différent car son champ d'action se veut plus large que celui d'un outil pour l'observation. Cependant jusqu'à maintenant, à l'exception des travaux de Dourish, sur lesquels nous reviendrons à la fin de cet article, l'ethnométhodologie a trouvé sa place uniquement dans les phases préliminaires de la conception, l'analyse de

¹ dans le domaine de la gestion des flux de travail, comme le workflow, ou dans le domaine des systèmes interactifs.

l'existant et des besoins, et dans les dernières phases, l'évaluation, démontrant presque toujours des difficultés d'usage et d'appropriation.

2.2 Vers la co-évolution dans les Systèmes Interactifs

Notre vision est que le SIE produit doit pouvoir s'adapter également aux besoins émergents, à la transformation des besoins ayant servis à l'analyse initiale, ainsi qu'aux évolutions des contextes d'usage et des technologies utilisées. Ces SIE peuvent être vus à double titre comme une intermédiation (un passage plus qu'une séparation) entre les concepteurs et les usagers :

- Une intermédiation entre les besoins, compétences, désirs... des usagers, les contraintes (technologiques ou autres), les compétences et les désirs des concepteurs. Le système est une inscription dans la technologie qui traduit l'équilibre local, temporaire, négocié, fragile entre les parties ;
- Une intermédiation entre les deux parties, usagers et concepteurs, pour en assurer la maintenance et l'évolution dans la durée. Il est clair que dans la plupart des approches ou méthodes de développement classiques des IHM, l'utilisateur est laissé livré à lui-même pour tout ce qui suit la phase de tests et de déploiement (Wild, Macredie, 2000). Il est donc nécessaire que le système interactif initialement développé puisse lui-même servir comme un pont entre les deux parties.

Nous pensons, utilisant pour cela l'expérience acquise au sein de la communauté du TCAO, que cette transformation continue possède une dimension collective et historique. C'est au sein de la communauté de pratique, au sens de la sociologie des usages, que les artefacts informatiques proposés doivent être adaptés afin de refléter la cristallisation des compétences et des attitudes de ses membres. C'est ce qui nous a amenés à nous tourner du point de vue théorique vers la Théorie de l'Activité.

Nous avons choisi le terme *co-évolution* pour traduire le fait que les SIE doivent être continuellement évolutifs, sans l'être toutefois de façon autonome ou auto-adaptative, car ils doivent rendre compte, de manière consciente, des évolutions des besoins, des attitudes et des compétences des usagers, individuellement ou collectivement. Mais il n'y a pas que le système qui doit évoluer. L'utilisateur doit également transformer, adapter ses pratiques et ses méthodes de travail, pour satisfaire les besoins évolutifs de l'organisation (Vicente, 2000). C'est une vue plus large que la boucle « tâche-artefact » proposée par Carroll et Rosson (1992) pour essayer de traduire la relation dialectique qui existait entre la spécification des besoins au travers de la tâche et l'artefact résultant du processus de conception. La co-évolution traduit tant les ajustements continus, négociés et socialement situés des pratiques de la part des individus, que les ajustements apportés au comportement du système interactif considéré. Ceci peut être synthétisé par le schéma visible en Figure 1, où il apparaît que l'environnement global fixe le contexte de la tâche (Kirsh, 1999).

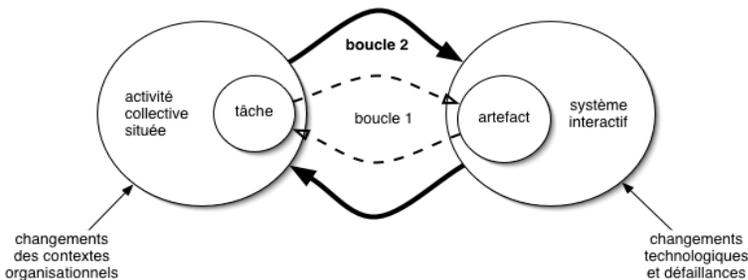


Figure 1. Illustration de la co-évolution.

3 La Théorie de l'Activité et son usage dans la conception des systèmes d'information

La Théorie de l'Activité (TA), telle que présentée par Nardi (1996), est un des courants, pour partie, issus des sciences sociales et ayant obtenu une large audience dans le domaine de la recherche en IHM ou en TCAO. La TA a d'abord été reconnue comme un domaine actif de recherche grâce aux contributions de l'école scandinave : Engeström (1987), Kuutti (1991 a & b, 1993, 1996) et Bodker (1991). Par exemple pour Kuutti (1991a) le concept d'activité est vu comme l'unité de base d'analyse pour les recherches sur le TCAO. Il faut rappeler que dans la conception des IHM pour l'utilisabilité, c'est le concept de tâche qui joue ce rôle central (Wild, Macredie, 2000). Nous verrons que nous allons re-questionner le rapport tâche-activité, tel que les IHM le conçoivent. Nous sommes conscients que ce cadre théorique est encore lui-même en évolution, et que sa complexité ne nous permet pas de prendre en compte actuellement toutes les contributions qu'il serait capable de fournir pour les problèmes posés par la production de SIE. Cependant, la TA offre un cadre théorique riche qui répond aux motivations des SIE tels que nous les envisageons. Il est également évident que notre travail pourrait être rapproché des théories dites instrumentales de la psychologie et de l'ergonomie contemporaines telles que P. Rabardel et ses collègues les ont développées (Rabardel, 1995). Mais cette proximité conceptuelle n'a pas pour l'instant fait l'objet d'un rapprochement plus systématique.

3.1 Histoire de la TA

La Théorie de l'Activité (TA) prend ses racines dans l'école historique et culturelle soviétique de la psychologie, fondée par Vygotski et ses disciples, notamment Leontiev et Luria. Vygotski s'est principalement focalisé sur la médiation par le langage et les processus d'apprentissage précoce chez l'enfant (Vygotski, 1997) (Clot, 1999). Sous l'influence de l'école scandinave d'abord, puis des chercheurs nord-américains, la Théorie de l'Activité² (encore appelée *Cultural Historical Activity Theory* ou CHAT si la dimension culturelle est plus importante) s'est concentrée sur la médiation par l'outil ou instrument, notamment dans les domaines de la technologie de la communication et de l'information. Progressivement, la TA s'est révélée être un corps de concepts dont le but est d'unifier la compréhension de l'activité humaine en fournissant des ponts vers les autres approches provenant des sciences humaines. En effet, les sciences du comportement ont toujours souffert d'une dichotomie existant entre l'individuel et le social. Si les sciences sociales utilisent le système social comme unité d'analyse, elles ont des difficultés à considérer l'humain lui-même. Les sciences du comportement, quant à elles, étudient les actions individuelles et ont des problèmes à en considérer le contexte. La solution offerte par la Théorie de l'Activité est la définition d'un concept intermédiaire, un contexte minimal pour l'étude des actions individuelles, qui sera considéré comme unité de base d'analyse : le concept d'**activité**. Plus précisément, la TA est un cadre conceptuel pour étudier différentes formes de pratiques humaines en tant que processus développementaux, combinant à

² Il faut noter que notre perception de la Théorie de l'Activité, telle que nous la présenterons dans la suite de l'article, est parcillaire au sens où, d'une part, tous les concepts qui font partie du cadre théorique ne seront pas mobilisés dans notre approche, et d'autre part, nous nous référons à un point de vue sur la TA, illustré par Nardi (1996), qui est fortement inspiré des dérivations faites par rapport à la pensée de Vygotski successivement par Leontiev puis Engeström. Ceci signifie que par rapport à d'autres points de vue sur cette Théorie de l'Activité il y a peut-être des divergences dans l'interprétation de l'œuvre séminale de Vygotski.

la fois et en même temps les niveaux individuels et sociaux. L'unité fondamentale d'analyse de la TA est l'activité humaine qui est définie comme un système cohérent de processus mentaux internes, d'un comportement externe et de processus motivationnels qui sont combinés et dirigés pour réaliser des buts conscients.

3.2 Les niveaux d'une activité

L'activité peut être décrite par une structure hiérarchique comprenant 3 niveaux : l'activité, l'action, et l'opération. L'activité (niveau le plus haut) est réalisée au travers de chaînes d'actions (niveau intermédiaire) elles-mêmes réalisées au travers d'opérations (niveau le plus bas).

Les actions peuvent être individuelles ou collectives, mais sont toujours dirigées vers un but conscient (intermédiaire par rapport à ce qui motive l'activité). Il est difficile, voire impossible, de comprendre une action en la considérant en dehors de son contexte, c'est-à-dire de son activité. Cependant, une action est souvent polymotivée, c'est-à-dire qu'une seule et même action peut appartenir à plusieurs activités. De ce fait, l'action effectuée peut être interprétée différemment suivant l'activité dans laquelle on la considère. L'exécution d'une action est planifiée en utilisant un modèle au cours d'une phase appelée orientation. Plus le modèle mis en œuvre est fiable, plus l'action a des chances d'atteindre son but. Dans le cas contraire, le sujet entre dans une phase d'analyse et d'apprentissage au cours de laquelle il sera certainement amené à modifier le modèle défaillant et, ainsi, augmenter son **expérience**.

Les opérations sont exécutées inconsciemment et sont orientées par une base d'orientation qui est établie au travers de l'expérience apparue au contact des conditions matérielles concrètes de l'opération (Bardram, 1997). Les opérations correspondent à des actions dont le modèle s'est démontré réellement fiable en fonction de certaines conditions. On peut considérer une opération comme une action qui est descendue d'un niveau dans la structure hiérarchique de l'activité, du fait de la quasi-disparition de sa phase d'orientation. Elle est alors exécutée plus rapidement et répond, de façon inconsciente, à des conditions spécifiques. Les actions opérationnalisées sont aptes à participer à la création de nouvelles et plus amples actions. Il arrive souvent, lorsque les conditions d'exécution d'une opération changent, que l'opération retourne au niveau de l'action, demandant une ré-analyse de la situation et une nouvelle phase d'orientation.

Comme le décrit Kuutti (1996), une action peut elle-même monter d'un niveau et devenir une activité. Symétriquement, une activité peut correspondre à une action dans une activité d'un niveau plus général. Ainsi, dans la TA, les limites entre les différents niveaux de la structure hiérarchique d'une activité sont mouvantes et floues. Ceci est particulièrement remarquable entre les niveaux activité et action car la classification dépend totalement du point de vue adopté.

La dynamique action-opération et la possibilité de construire, au fil du temps, des actions de plus haut niveau à partir d'actions opérationnalisées constitue une propriété fondamentale du développement humain. Ces mécanismes nous paraissent importants car ils viennent contribuer à la formation de l'expérience des sujets.

3.3 Les propriétés de l'activité

La TA définit huit propriétés de base du concept d'activité :

1) Une activité possède un objet (matériel brut ou espace problème) vers lequel elle est dirigée, qui la différencie d'une autre activité, et qui en motive l'existence. La réalisation de l'objet produit une transformation du monde comme, par exemple, la création d'un artefact.

2) Une activité possède au moins un sujet (acteur) actif qui comprend et est en accord avec son objet. Tous les participants d'une activité ne comprennent pas ou ne

reconnaissent pas forcément l'existence de l'objet de l'activité, auquel cas ils sont identifiés comme des sujets passifs. Le sujet peut être individuel ou collectif.

3) Une activité est un phénomène collectif. Comme nous venons de le voir dans le point précédent, le sujet peut être lui-même un collectif. De plus, le sujet fait généralement partie d'une communauté, c'est-à-dire d'un ensemble de sujets partageant le même objet d'activité. Enfin, une activité est toujours influencée par ses conditions environnementales et le sujet impliqué dans une activité, même s'il est seul, est toujours plus ou moins directement influencé par d'autres sujets impliqués dans des activités connexes.

4) Une activité est réalisée par ses participants au travers d'actions orientées vers des buts conscients.

5) Une activité existe dans un environnement matériel qu'elle transforme.

6) Une activité est un phénomène de développement historique.

7) Les contradictions qui y apparaissent en sont la force de développement.

8) Les relations existantes entre les éléments d'une activité sont médiatisées culturellement.

Notre but étant de fournir un support à l'activité coopérative, certains de ces points ont déjà une résonance particulière pour notre conception : l'activité transforme son environnement, c'est un phénomène dynamique, de développement historique et empli de contradictions.

3.4 La structure de base d'une activité

Engeström (1987) a défini un modèle structurel simple du concept d'activité intitulé « structure de base d'une activité » qui constitue la référence la plus classique à la TA. Ce modèle exprime la relation entre le *sujet* et l'*objet* de l'activité. Cette relation est réciproque : le sujet réalise l'objet de l'activité, mais en même temps, les propriétés de l'objet transforment le sujet en augmentant son expérience. Elle est de plus médiatisée par le concept d'*outil* représentant tout ce qui est utilisé dans le processus de réalisation de l'objet, aussi appelé transformation de l'objet, incluant aussi bien les outils matériels que les outils pour penser. L'outil à la fois permet et limite : il permet au sujet de réaliser l'objet de son activité, mais limite en masquant une partie du potentiel de transformation. De plus, Bertelsen (1998) ajoute que l'outil informe sur l'état de l'objet en cours de transformation. Les outils supportent et complètent les capacités humaines dans la construction de systèmes plus efficaces, permettant des accomplissements plus hauts. En contrepartie, l'outil participe à la formation des objectifs des sujets qui l'utilisent car il porte en lui des buts implicites qui y ont été mis par ses développeurs. D'ailleurs, Christiansen (1996) rappelle qu'un outil doit servir le double but de permettre de faire quelque chose et de rappeler que quelque chose doit être fait. D'un autre côté, l'outil est lui-même transformé et construit au cours de l'activité. Il est souvent modifié ou adapté par les sujets, en fonction de leurs besoins émergents, de leurs buts, et de leur expérience. Ainsi, cette expérience se retrouve dans l'outil qui porte en lui l'héritage culturel de la situation. La médiation par l'outil correspond alors à un moyen de transmettre une certaine culture. La structure de base d'une activité souligne aussi que l'individu n'est pas isolé mais fait partie d'une *communauté* qui partage le même objet d'activité. Les relations *communauté-sujet* et *communauté-objet* sont médiatisées par les concepts de *règles* et de *division du travail* contenant eux aussi l'héritage culturel de la situation. À l'instar de l'outil, ces médiateurs sont ouverts à de nouveaux développements.

3.5 Les apports de Kuutti : expansivité, organisation

En 1991, Kuutti (1991a) explique que le nouveau champ de recherche que constitue le TCAO souffre du problème de la sélection d'une unité de base

structurelle et fonctionnelle pour analyser le travail. Selon lui le TCAO peut être défini comme le travail de multiples sujets actifs qui partagent un objet commun et supporté par la technologie de l'information. Il souligne que l'élément clé de cette définition est le concept de sujets actifs qui permet de différencier le TCAO des systèmes d'information traditionnels dans lesquels la prédétermination de séquences de travail par le système correspond au cas normal d'exécution. Ainsi, il définit l'activité comme unité de base d'analyse pour la recherche en TCAO.

La seconde contribution de Kuutti (1991 a & b) qui nous semble très importante consiste en une classification des types de base de support au travail, du point de vue de la technologie de l'information : une première catégorie relative aux systèmes d'information traditionnels adoptant une approche Tayloriste du travail, une seconde catégorie correspondant aux systèmes « classiques » de support au TCAO, c'est-à-dire ceux mettant en jeu des sujets actifs partageant un objet commun, et une troisième catégorie décrivant une approche hors normes dans la recherche en TCAO. Cette approche, que nous avons fait nôtre, définit le système ultime de TCAO comme celui qui supporterait l'expansivité de l'activité, c'est-à-dire qui permettrait la reconstruction du travail en créant les artefacts informatiques nécessaires, cette reconstruction étant réalisée par les travailleurs eux-mêmes.

La dernière contribution de Kuutti (1993) que nous présentons utilise la TA pour replacer les activités dans une analyse organisationnelle. Dans les situations réelles, il existe toujours un réseau d'activités interconnectées. Kuutti rappelle que la participation à diverses activités interconnectées, possédant des motifs très différents, peut causer des tensions et des distorsions. Les connexions peuvent former un réseau et, puisqu'une activité délimite une partie des processus de travail dans une organisation, ce concept peut être utilisé pour analyser les organisations. Kuutti décrit un ensemble non-exhaustif de types de connexions. Le principe de base est que chacun des éléments d'une activité peut constituer l'objet d'une ou de plusieurs autres activités dans l'organisation.

3.6 Les apports de Bardram : coordination, coopération et co-construction

Dans une démarche proche de celle de Kuutti avec sa proposition pour un support à l'expansivité, Bardram (1997) souligne que l'anticipation guide l'activité en créant une synthèse entre la perception de l'état environnemental (la situation) et l'expérience du sujet. Cependant, la situation et l'expérience du sujet ne cessent d'évoluer au cours de l'activité. C'est pourquoi une anticipation n'est réellement valide qu'au moment de sa construction. Les plans d'action sont des artefacts cognitifs ou matériels qui supportent l'anticipation et qui médiatisent l'activité. Comme tout élément médiateur, le plan d'action doit donc être révisable in situ. Les pannes qui peuvent survenir dans un plan d'action ne doivent pas être considérées comme des exceptions dans l'activité, mais comme des parties naturelles et très importantes. Elles forment la base de l'apprentissage et donc du développement et de l'amélioration des plans pour les actions futures. Bardram souligne d'ailleurs le côté réflexif de l'activité : le processus d'apprentissage crée et améliore le plan qui, à l'origine, guidait l'activité.

Ainsi, dans le travail, les routines, la coopération et les conflits ne doivent pas être vus comme des catégories distinctes, mais comme différents aspects du même effort. Bardram définit alors ce que sont la coordination, la coopération et la co-construction de l'activité. La coordination correspond au niveau le plus bas où les individus ne font que se concentrer sur la réalisation d'actions prédéterminées par un script d'exécution. La coopération, niveau intermédiaire, est celui où les sujets sont actifs. La co-construction représente les interactions dans lesquelles les sujets re-conceptualisent leur organisation et leurs interactions en relation à l'objet partagé.

Bardram propose alors deux principales propriétés que doivent posséder les systèmes de support au TCAO et auxquelles nous adhérons : (1) supporter les activités de travail collaboratif dans les trois niveaux, ainsi que les transitions entre ceux-ci, et (2) intégrer ce support à la coopération dynamique dans le support de l'objet du travail lui-même. En d'autres termes, les coordination, coopération et co-construction du travail doivent être supportées par le système, et ces fonctionnalités doivent être, du point de vue du sujet (et pas seulement du développeur), intimement liées à l'activité supportée.

3.7 L'utilisation de la Théorie de l'Activité dans la conception des SIE

Dans la quasi-totalité des utilisations de la TA, les éléments d'une situation réelle instrumentée sont typés et replacés dans le cadre fourni par la TA. Les auteurs étudient alors les relations qui existent entre ces éléments, en recherchant les contradictions qui vont leur permettre de générer une nouvelle situation, en modifiant de manière ad hoc le système étudié. Même s'ils sont intéressants, ces résultats sont fortement liés à la situation spécifique étudiée, et donc difficilement réutilisables d'une manière générale.

D'autres travaux tentent de réifier ces résultats et fournissent des lignes directrices générales sous forme de check-lists (Kaptelinin et al., 1999). Malheureusement, ces check-lists sont difficiles à utiliser pour les informaticiens concepteurs que nous sommes. Elles fournissent un guide permettant d'analyser une situation, mais ne sont pas directement reliées aux propriétés intrinsèques du système en cours de création.

Ces approches utilisent la structure de base de l'activité et en amont ou en aval des processus de conception des systèmes technologiques étudiés. Dans nos infrastructures informatiques, DARE et CoolDA, la partie la plus importante provenant de la TA n'est pas l'identification qu'une activité est constituée de sujets, d'outils, de règles, etc., mais l'explicitation des mécanismes qui sont liés à leur développement. Ainsi, ce qui nous intéresse le plus n'est pas la description des éléments constituant l'activité humaine, mais bien un point de vue générique sur les dynamiques qui se créent autour d'eux. À notre connaissance, le seul exemple de travaux proches des nôtres de ce point de vue sont ceux de Kaptelinin (2003) dans UMEA. Kaptelinin souligne que les environnements informatiques actuels ne sont pas conçus pour gérer des tâches de haut niveau. Cette idée est très proche de la nôtre tentant de fournir avec DARE et CoolDA un environnement d'articulation de divers collecticiels participant à des tâches de haut niveau par rapport aux tâches spécifiques pour lesquels ils sont conçus. Kaptelinin tente de transformer les métaphores de bureau, de répertoires, ou encore de systèmes de gestion d'informations personnelles comme Outlook pour les repenser dans le cadre de la TA. Même s'il possède des limites, la force d'UMEA est qu'il capture les actions de l'utilisateur au sein de ses activités, de manière à conserver leur situation et l'historique du développement de l'activité. Comme nous, Kaptelinin s'appuie sur le concept d'activité pour créer un environnement global et met ici plus en avant les mécanismes développementaux exposés dans la TA que le simple nom des entités qui la composent. La différence majeure entre ces travaux et les nôtres est le fait que nous nous intéressons à la création d'un environnement distribué principalement destiné à supporter la coopération, alors qu'UMEA est moins porté sur la dimension coopérative de l'activité.

Ainsi, comme nous l'avons souligné dans la partie 2.1, et y compris pour l'utilisation de la TA, les approches classiques (cf. Figure 2) sont caractérisées par des contributions des SHS et de l'Informatique qui sont bien distinctes et présentent une faible interrelation. Par contre dans l'approche que nous proposons, il y a la volonté

d'intégrer les deux dimensions, humaines et technologiques, et de les fondre dans un même cadre théorique.

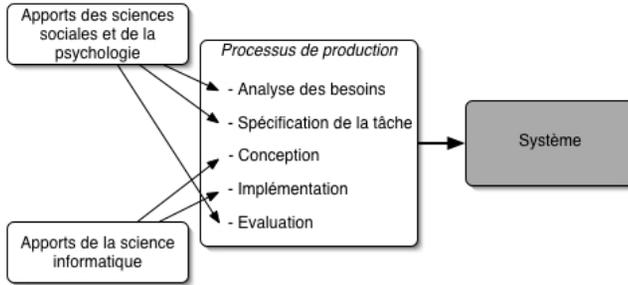


Figure 2. Les approches classiques d'intégration des SHS.

Cette approche de la conception des futurs Systèmes Interactifs et Evolutifs est présentée dans la Figure 3. L'adjonction d'un méta modèle et d'une infrastructure constitue un cadre de conception pour la génération et la gestion des changements du modèle conceptuel propre à un SIE donné. Ce cadre va supporter un comportement réflexif car il pourra être interrogé par une instance particulière de SIE afin de connaître son propre comportement (l'introspection) et offrir un contexte clair et un support aux transformations que ce système devra opérer sur lui-même (l'intercession) pour satisfaire aux évolutions des tâches et de leurs contextes. Le besoin de faire appel à la réflexivité des systèmes est en accord avec notre argumentation développée au début de ce papier, avec les propositions de Kuutti et Bardram. Il est lié aux contraintes découlant de la nécessité de gérer cela au cœur même du système : pour et par le système.

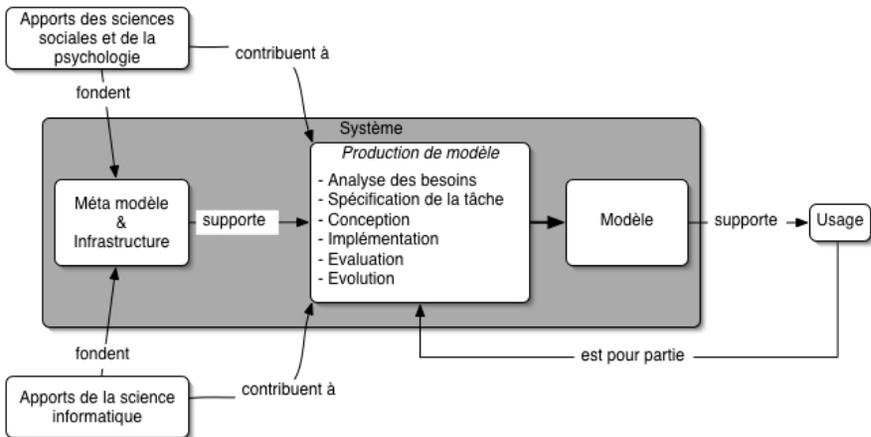


Figure 3. L'introduction d'un méta modèle et d'une infrastructure qui intègrent des apports de la science informatique et des fondations venant des sciences humaines et sociales.

Nous avons choisi de réaliser cette approche dans une méta architecture, au sein de DARE & CoolDA, qui vise à offrir un support logiciel générique permettant à différents SIE spécifiques de trouver un milieu favorable à la gestion de leurs évolutions, ceci dans une perspective psycho-socio-historique. Ainsi, notre objectif

principal n'est pas de développer un tableau blanc partagé, un outil de chat ou de mail par exemple, mais de fournir un environnement qui intègre et articule l'utilisation de ces outils. Il existe d'autres travaux de recherche sur la création de collectifs malléables et intégrateurs de composants (Hummes et al., 1998) (Roussev et al., 2000). Cependant, pratiquement tous sont uniquement influencés par les apports de la science du logiciel en matière de « composabilité » et d'interopérabilité des différents éléments pouvant être mobilisés au sein d'une même infrastructure supportant une activité collaborative, sans prendre en considération le contexte local d'intégration. Les infrastructures informatiques DARE et CoolDA sont fortement inspirées des travaux liés à la TA, c'est pourquoi nous nommons ce contexte d'intégration un *Support d'Activité* (SA).

4 De DARE à CoolDA : les évolutions de notre conception d'une infrastructure informatique fondée sur la Théorie de l'Activité

4.1 Le Support d'Activité

Comme nous venons de le voir, le concept de Support d'Activité (SA) est central dans notre approche. Un SA permet aux utilisateurs d'accéder à leurs ressources (coopératives ou non) en fonction de leur rôle. Chaque SA correspond au support informatique offert aux utilisateurs impliqués dans une activité spécifique. La définition d'un SA se réfère à un ensemble de concepts inspirés de la TA. Ces concepts sont représentés dans la Figure 4 qui constitue notre modèle ontologique.

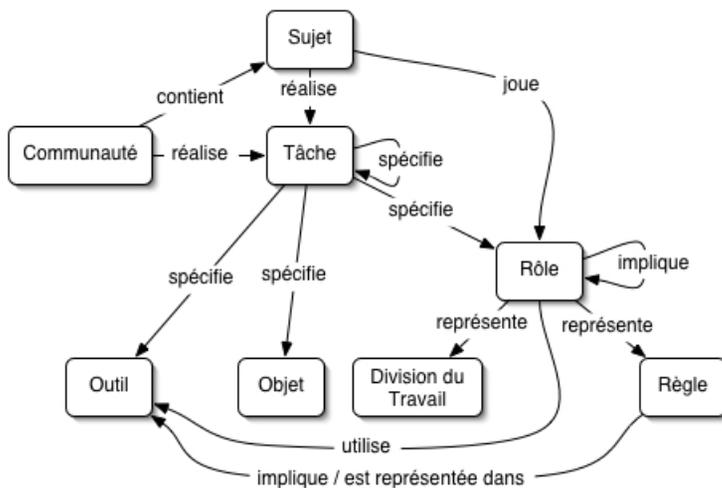


Figure 4. Éléments d'un modèle ontologique fondé sur la TA.

Dans un SA, on trouve un ensemble de sujets faisant partie d'une communauté impliquée dans la réalisation d'une tâche. La notion de tâche s'écarte un peu de la vision connue en ergonomie du logiciel et correspond ici à la définition fournie par Leontiev qui a été reprise par Bedny et Meister (1997): « *Task is a situation requiring realisation of a goal in specific conditions* ». C'est la réalisation de la tâche qui constitue l'activité supportée par le système. Ainsi, dans DARE et CoolDA, la tâche spécifie l'objet à réaliser, ainsi que l'ensemble des outils et rôles qui sont mis en œuvre dans l'activité. Le rôle représente une partie de la division du travail et des règles qui existent dans l'activité réelle supportée par le système. Il définit la manière dont un

sujet accède aux outils permettant d'agir dans l'activité. Rappelons que, au-delà du rôle, chaque outil à la fois permet et limite les actions des sujets. Ceci est d'autant plus vrai dans les situations médiatisées par l'ordinateur qui nous intéressent ici puisque chaque outil peut lui-même implémenter une partie des règles qui régissent l'activité. Par exemple, l'utilisation d'un outil de messagerie instantanée avec une certaine configuration peut interdire (dans une certaine mesure) ou tout au moins influencer les sujets pour qu'ils ne puissent discuter en aparté, ce qui peut correspondre à une règle régissant leur communauté.

Dans notre première approche, DARE, et d'une manière classique pour le domaine des IHM, la tâche pouvait se décomposer en sous-tâches (Bourguin, Derycke, 2000). Ici, tel que nous le présentons dans la Figure 4, la notion de sous-tâche semble avoir disparu³. Cette notion peut cependant être rapprochée du lien qui lie l'entité *Tâche* à elle-même. Toutefois, en fonction des problèmes et des évolutions qui seront présentés plus en détail dans les parties 4.3 et 4.4, nous préférons aujourd'hui considérer qu'une tâche peut spécifier l'implication d'une autre tâche. La notion de sous-tâche implique une dimension hiérarchique qui nous semble parfois inappropriée en fonction du point de vue adopté. Par exemple, une tâche A peut très bien exister et avoir pour but la production d'un artefact qui se révèle utile à la réalisation d'une tâche B en cours de définition. A peut être considérée comme une sous-tâche du point de vue de B, mais n'a pas été créée par cette dernière. Le lien *spécifie* permet de considérer des réseaux d'activité plutôt que des hiérarchies. Ce réseau de tâches (ou d'activités lors de leur réalisation) est directement à rapprocher au lien qui lie l'entité *Rôle* à elle-même. Ce lien signifie qu'un rôle dans la réalisation d'une tâche peut impliquer un autre rôle dans la réalisation d'une autre tâche.

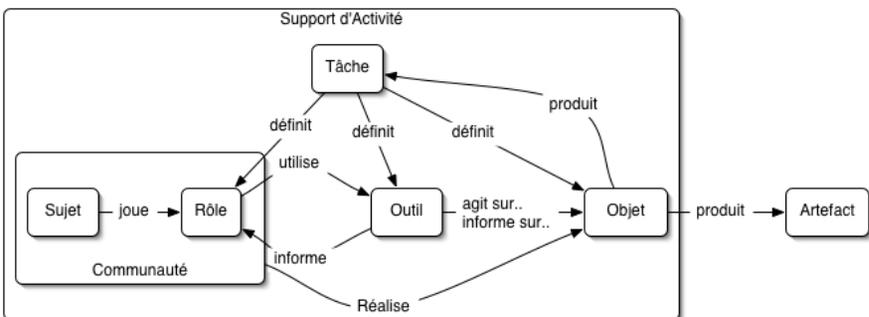


Figure 5. Le support d'activité et sa tâche : un système réflexif.

La Figure 5 présente le modèle conceptuel de ce que nous avons appelé SA. Ce modèle met en œuvre les concepts que nous venons de définir. Il rappelle aussi le fait que l'outil permet et limite la réalisation de l'objet de l'activité et informe le sujet, en fonction de son rôle, sur l'état de sa réalisation. De plus, ce modèle introduit des mécanismes qui permettent de supporter les propriétés réflexives de l'activité : pour supporter l'expansivité, nous rendons la tâche accessible au sein du SA. Cet accès est rendu possible grâce à des outils que nous nommons *outils méta* du fait qu'ils permettent d'accéder à la définition d'une activité, c'est-à-dire à la spécification d'une tâche. Les outils méta permettent à la communauté d'être informée sur la tâche du

³ Il en est de même pour la notion de micro-rôle qui existait entre le rôle et l'outil. Cette notion n'a plus de raison d'exister du fait des changements apportés aux modèles présentés dans la partie 4.4 de l'article.

système et, éventuellement, de la transformer si elle est en contradiction avec ses besoins. L'utilisation d'un outil méta correspond à une *méta activité* par rapport à l'activité de base. Par exemple, dans le cas d'une activité de base dont l'objet est de produire un article de recherche, la réflexion sur la manière de produire cet article correspond à une méta activité dont l'objet est de produire la situation pour réaliser l'activité de base. En d'autres termes, une partie de l'objet de l'activité du système peut être de produire sa propre tâche. Une relation causale entre le SA et sa tâche implique que toute modification de la tâche entraîne des répercussions directes sur le SA. Ainsi, la tâche définit un SA qui est capable d'y accéder en lecture et/ou modification, ce qui définit un système réflexif. Il faut aussi remarquer que dans ce modèle de SA, un artefact produit de la réalisation de l'objet peut être transmis à l'extérieur, à une autre activité. Si l'on revient aux sources de la TA, on peut considérer que cette « production » de l'activité peut correspondre à des artefacts de niveaux différents tels que Wartovsky et Engeström les décrivent (Engeström, 1987). Ceci est cohérent avec la réflexivité introduite dans la Figure 5 car l'artefact produit peut être un plan d'activité, c'est-à-dire la spécification d'une tâche, un artefact secondaire qui peut être consommé par l'activité qui l'a généré.

4.2 Réflexions sur la réflexivité

Une approche originale de la réflexivité : la co-évolution

La réflexivité est une propriété qui a été déjà identifiée par d'autres chercheurs dans le domaine du TCAO comme étant co-substantielle des activités coopératives (Dourish, 1998), y compris dans la communauté française (Saikali, David, 2001) (Ferraris et al., 2002). La littérature fait apparaître différentes approches de la réflexivité, mais la plupart ne la considèrent que du point de vue des développeurs. Dans ce cas, l'environnement est malléable car ses concepteurs peuvent accéder et modifier sa définition. Cette malléabilité du système est en général souhaitée comme accessible au cours de sa propre exécution. Mais on est plus proche de mécanismes de migration planifiée (anticipation dans la conception) que de véritable co-évolution dans l'usage (Schuurman, Heeren, 1999). Il est donc préférable de s'inscrire dans une approche qui tente d'aller plus loin et qui veut fournir un accès à ces mécanismes réflexifs aux utilisateurs finaux. Ainsi, la malléabilité sera directement accessible à ceux qui en ressentent le besoin, *in situ*.

Les systèmes de TCAO possédant des propriétés réflexives qui sont offertes aux utilisateurs finaux, comme dans (Ferraris et al., 2002), les intègrent souvent comme des fonctionnalités supplémentaires données à un type prédéfini d'utilisateurs du système. Il s'agit par exemple du rôle classique et stéréotypé d'*Administrateur*. Dans cette approche, le modèle d'exécution du collecticiel est réifié. Dans DARE et CoolDA, cette réification est réalisée par le biais de l'entité *Tâche* et de ses composants (*Tâche, Objet, Rôle, Outil*). Ce type d'utilisateur peut ainsi redéfinir dynamiquement les jeux de rôles qui régissent les activités coopératives. Toutefois, DARE et CoolDA se distinguent encore de ces collecticiels réflexifs en voulant de plus supporter la co-évolution, avec sa dynamique historique et culturelle.

La co-évolution correspond à une approche plus globale et coopérative de la malléabilité. La redéfinition du système par ses utilisateurs est considérée comme une partie de l'activité coopérative supportée par le système et nommée méta activité. Dans le modèle conceptuel de SA que nous venons d'exposer (cf. Figure 5), il est à noter que les outils méta sont gérés comme des outils à part entière. Les acteurs peuvent alors redéfinir coopérativement l'environnement de coopération en fonction de leurs rôles. En d'autres termes, la méta activité est intimement liée à l'activité

coopérative, elle est elle-même coopérative, et le système se considère lui-même comme un artefact médiateur de l'activité coopérative qu'il supporte.

Une telle approche tente clairement de répondre aux besoins exposés par Kuutti et Bardram à propos du système ultime de TCAO supportant à la fois la coordination, la coopération et la co-construction au sein du système. La coordination et la coopération sont supportées au sein du SA. La méta activité a pour but de supporter l'expansivité et la co-construction le système.

La cristallisation de l'expérience

La propriété de cristallisation et réutilisation de l'expérience identifiée dans la TA est aussi un mécanisme qui nous semble fort important et qui distingue notre approche. La TA souligne qu'un artefact transformé au cours des activités par ses utilisateurs cristallise leur expérience. Au-delà de la mise en place de mécanismes réflexifs, nous mettons donc aussi l'accent sur ce mécanisme de cristallisation et voulons mettre à profit l'expérience des utilisateurs de nos infrastructures informatiques. Une telle expérience est par exemple la spécification d'un jeu de rôles pour un type d'activité particulière. Si ce jeu de rôle évolue au cours d'une activité, du fait de l'expérience des acteurs, nous voulons qu'il soit accessible pour d'autres activités, à ceux qui désirent le mettre en œuvre et qui le feront peut-être eux-mêmes évoluer.

Dans notre approche, la tâche est utilisée pour instancier des SA. Le système prenant en compte la notion de méta activité au cours de laquelle une tâche peut évoluer, la tâche constitue l'élément qui cristallise et permet la réutilisation de l'expérience des sujets envers le système. Le support de ce mécanisme dans nos travaux est synthétisé dans la Figure 6. Une communauté réalise une activité dans le monde réel. Cette activité est supportée par le système. La tâche système correspond à la partie de la tâche réelle de la communauté qui a été explicitée et spécifiée au sein du système pour qu'il puisse la supporter. La communauté acquiert une certaine expérience envers son activité au fur et à mesure qu'elle la réalise. Cette expérience peut être rendue explicite grâce et au sein du système en modifiant sa tâche. Par exemple, l'évolution de la division du travail dans l'activité peut produire un nouveau jeu de rôles qui sera spécifié dans la tâche système. Cette nouvelle tâche correspond à un nouveau modèle de support d'activité qui peut être mis en œuvre par une autre communauté aux besoins similaires.

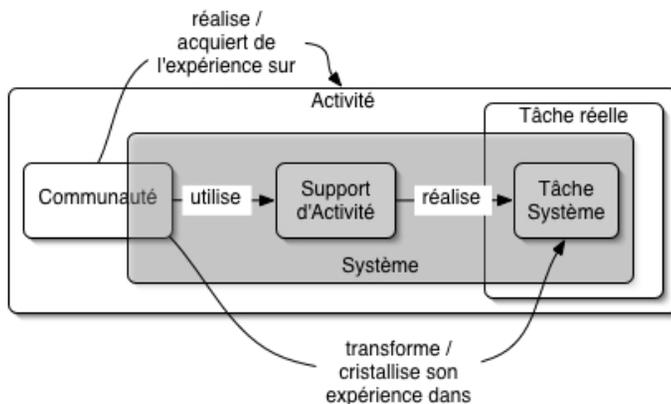


Figure 6. *La cristallisation de l'expérience.*

4.3 Les leçons du premier projet : DARE

Les principes et modèles que nous venons d'exposer sont ceux qui ont guidé nos travaux depuis la genèse du projet DARE, première tentative de conception d'une infrastructure fondée sur la TA. Cependant, dans DARE, ces principes et modèles n'avaient pas encore la forme qu'ils ont dans cette présentation. Cette première conception, et son développement informatique, nous ont donc servis de laboratoire expérimental pour affiner, tester et mettre à l'épreuve les idées que nous avons pour une telle approche.

DARE en tant que laboratoire expérimental

Il nous faut d'abord préciser que nous n'avons jamais eu l'occasion de tester la plateforme réalisée dans DARE avec des utilisateurs finaux dans des situations d'usage réel. Ceci provient du fait que DARE s'apparente plus à une infrastructure informatique qu'à une application particulière. Le but de DARE, de même que celui de CoolDA aujourd'hui, n'est pas la création d'une application de TCAO, mais bien de fournir une base à la création et l'évolution de diverses applications de TCAO, ou SA, réalisée en articulant divers collecticiels existants. Or, la manière dont les collecticiels sont aujourd'hui conçus rend difficile leur intégration. Ces applications ne sont pas encore pensées comme des composants de haut niveau qui servent des activités plus globales que celle qu'elles supportent directement. Par exemple, un outil de tableau blanc partagé sert en général une activité plus globale que celle du dessin collaboratif. Il s'agit par exemple d'une activité de conception dans laquelle le tableau sera utilisé en conjonction avec une audioconférence, un gestionnaire de versions, etc. Une plus ample description de ces problèmes peut être trouvée dans (Bourguin, 2004). De ce fait, nous nous situons directement dans la problématique soulignée par Edward et al. (2003) qui posent la question de l'évaluation des infrastructures. Ces auteurs insistent sur les bienfaits de la création d'infrastructures du point de vue de la généralité des concepts et mécanismes qu'elles supportent, mais aussi sur la difficulté liée à leur évaluation. Les infrastructures ne sont en effet pas directement des applications utilisables en situation concrète, mais elles possèdent des propriétés qui vont fortement influencer les applications qui seront créées grâce à elles. Ceci est d'ailleurs bien une volonté dans nos travaux car nous voulons améliorer la conception des SIE en tentant de marier les apports des SHS et de l'Informatique au travers de leur inscription dans notre infrastructure. Diverses méthodes d'évaluation d'applications avec des utilisateurs finaux existent, mais l'évaluation d'infrastructures se fait généralement d'un point de vue technique avec des critères tels que la performance, la robustesse, le passage à l'échelle, etc. Dans DARE et CoolDA, nous avons toujours testé nos propositions du point de vue de la faisabilité technique. Malheureusement, de telles infrastructures ont aussi besoin d'informations sur l'utilisabilité qu'elles vont induire dans les applications supportées. Il n'existe pas à notre connaissance de solution bien définie à ce problème complexe : pour tester une infrastructure, il faudrait avoir implémenté des applications qui l'utilisent. Edward et al (2003) soulignent d'ailleurs que même lorsque de telles applications existent, il n'est pas aisé d'identifier les éléments qui permettent d'évaluer l'infrastructure elle-même, puisqu'elle est en général cachée par l'application. Mais pour implémenter des applications, il faut avoir réalisé l'infrastructure ! Devant ce dilemme, et avant d'être en mesure de tester dans des conditions réelles des instances de SA avec des utilisateurs finaux, nous nous basons sur les apports fournis par la TA et évaluons nos choix d'un point de vue théorique. C'est cette réflexion qui nous a amené à manipuler les concepts fournis par la théorie, à les confronter à des choix de

conception, à acquérir une certaine expérience et à la cristalliser dans la suite de DARE : l'infrastructure CoOLDA.

Les paragraphes qui suivent soulignent les changements les plus marquants, résultant de notre expérimentation, qui nous ont amenés à passer de DARE à CoOLDA. Ils soulignent les questions et embryons de réponses qui ont émergé de nos travaux tentant de mêler les fondements de l'activité humaine aux fondements des systèmes que nous créons pour la supporter.

La nécessité d'un modèle récursif sans décrochement

Dans la partie 4.1, nous avons vu que pour fournir un accès aux spécifications d'une activité de base, il faut fournir aux utilisateurs un outil qui leur permet d'accéder au niveau méta du système. Dans DARE, cet outil est ajouté dans l'activité de base. Nous pensons aujourd'hui que ce choix est néfaste au regard de la co-évolution.

Prenons l'exemple exposé dans la Figure 7. Dans le contexte d'une activité de *cours*, il peut être important pour un *enseignant* d'accéder à la redéfinition de sa séance. Dans DARE, il suffit alors de lui fournir un outil pour lequel il a des droits particuliers (au regard des étudiants par exemple). Le problème se pose si un *technicien* de l'environnement doit aussi avoir accès à ce niveau méta. L'inscription du technicien en tant que membre du cours serait artificielle : le technicien ne suit pas le cours. Il est possible de lui fournir l'accès à cet outil en l'ajoutant à son propre SA, supportant une activité différente du cours. Cependant, du point de vue de la co-évolution, l'enseignant doit pouvoir faire évoluer son environnement en fonction du déroulement de son activité, ceci dans un processus coopératif impliquant potentiellement aussi le technicien.

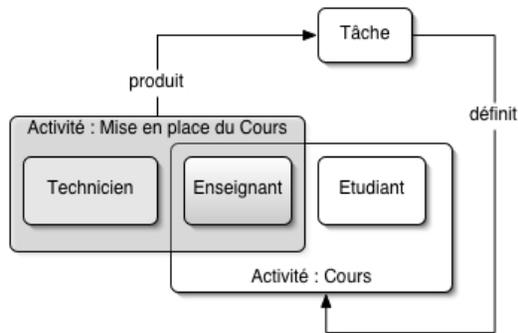


Figure 7. Exemple d'activités liées mais distinctes.

C'est pourquoi, nous sommes aujourd'hui convaincus que l'accès au niveau méta n'est pas le simple fait d'un ajout d'outil dans une activité. La méta activité n'est pas qu'une partie de l'activité de base, elle constitue une activité à part entière impliquant plusieurs acteurs provenant eux-mêmes d'activités différentes mais liées. Les travaux de Kuutti sur la TA et les organisations (cf. 3.5) nous confortent dans cette nouvelle approche. Kuutti souligne comment les sujets sont souvent impliqués dans différentes activités interconnectées. Le produit d'une activité peut constituer un des artefacts d'une autre activité. De notre point de vue, une méta activité a pour but la production (ou l'évolution) de tout ou partie des spécifications de son activité de base. Ainsi, les acteurs de la méta activité peuvent être impliqués dans l'activité de base, mais pas obligatoirement. Réciproquement, les acteurs de l'activité de base peuvent être impliqués dans la méta activité. Du fait de nos choix de conception, ce

n'était pas le cas dans DARE. Il est donc nécessaire que nous fassions évoluer nos choix pour que CoolDA possède des fondements plus aptes à supporter la co-évolution.

Les bienfaits de l'approche par modèles ?

Pour réaliser DARE, nous nous sommes fortement inspirés du Meta-Object Protocol ou MOP (Kiczales et al., 1991), une approche Orientée Objet des systèmes informatiques réflexifs. D'autres chercheurs ont d'ailleurs eux aussi utilisé cette approche pour réaliser plus aisément les dimensions réflexives de leur collecticiels (Dourish, 1998). Adaptant l'approche MOP, la réflexivité dans DARE est rendue possible grâce à l'introduction d'un méta modèle dont l'entité majeure est la *Tâche*. Chaque tâche est un modèle d'activité et il est possible de créer différentes instances de ce modèle.

Il est important de souligner que nous utilisons la tâche d'une manière différente que dans l'approche classique fortement critiquée ces dernières années, la modélisation de la tâche étant jugée trop rigide et peu favorable à la malléabilité souhaitée. Notre tâche est bien un modèle qui décrit comment doit se dérouler l'activité coopérative, mais ce modèle est évolutif ! Il décrit la manière dont le système supporte l'activité à un instant donné. Toute modification de la tâche entraîne des répercussions directes sur les activités correspondantes. Ainsi, notre approche de la tâche peut être comparée aux plans de Suchman (1987) et de Bardram (1997) : une tâche est nécessaire (en particulier dans un environnement informatisé qui supporte sa réalisation) mais révisable in situ.

L'approche par modèles est fort intéressante, en particulier du point de vue de la cristallisation de l'expérience. Lorsque les acteurs dans une activité accèdent à son niveau méta, ils modifient la tâche, c'est-à-dire un modèle dans lequel l'expérience alors cristallisée peut être directement mise à profit. Une des questions majeures est celle de la stabilisation du système dans le cadre des approches réflexives. Pour qu'une expérience naisse d'une activité, il est nécessaire que le système passe par des phases stables. Les acteurs ne pourront correctement comprendre et faire évoluer leur environnement que si celui-ci cesse temporairement de changer. Nous espérons que les dimensions humaines et coopératives de notre approche permettront de pallier ce problème : dans la co-évolution, toute évolution devrait être le résultat d'un processus coopératif. Ceci paraît réalisable dans le contexte d'activités fortement interconnectées. Le questionnement est plus grand pour l'évolution d'une tâche dont existent plusieurs activités disjointes, impliquant des utilisateurs différents. Puisqu'il existe une relation causale entre une tâche et ses activités-instances, les acteurs qui modifient leur tâche transforment aussi la spécification de toutes les autres activités issues de cette tâche. Cette problématique est symbolisée dans la Figure 8. Les activités A'1 et A'2 sont instances de la tâche T'. Lorsque l'activité A'1 modifie sa tâche (T''), par exemple en modifiant un rôle, elle modifie aussi le comportement de l'activité A'2. Se pose alors un dilemme : faut-il ou non faire profiter toutes les activités en cours ou futures d'une expérience développée dans une activité particulière similaire ? Nous pensons que la réponse à cette question dépend fortement du contexte d'utilisation du système et notre travail à l'heure actuelle consiste à découvrir les moyens de mettre en œuvre les deux options.

L'analogie avec des mécanismes des langages objet nous questionne aussi au niveau de la notion d'héritage. L'héritage est intéressant car il permet de définir une tâche à partir d'une autre. Par exemple, une tâche comme la *Rédaction d'un article* peut être spécialisée en *Rédaction d'un article pour la Revue d'Interaction Homme-Machine (R.I.H.M.)*. Comme dans le monde objet, c'est un moyen efficace pour réutiliser de l'expérience. Un autre point appréciable de l'héritage est qu'il facilite, d'un certain

point de vue, la gestion de l'évolution des modèles : la modification d'un modèle entraîne des modifications automatiques dans tous les modèles qui en héritent et il n'est pas nécessaire de les modifier explicitement. Même si ce mécanisme est intéressant, il nous questionne pour nos modèles d'activités : une tâche définie à partir d'une autre doit-elle toujours subir les répercussions des transformations de celle dont elle est issue ? Dans la Figure 8, la tâche T a pour instance l'activité A. La tâche T' est une spécialisation de la tâche T. Lorsque l'activité A modifie sa tâche (T), le lien de spécialisation influence aussi le comportement des activités A'1 et A'2. De la même manière que précédemment, nous pensons que les choix liés à la portée des modifications du système dépendent du contexte d'utilisation. Nous réfléchissons donc aux moyens qui permettent la mise en œuvre de la notion classique d'héritage, ainsi qu'aux mécanismes qui permettent de s'en détacher.

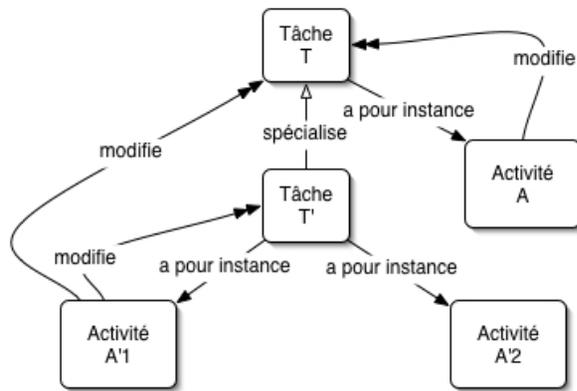


Figure 8. Les répercussions des modifications dans une approche par modèle et relation causale.

Un dernier point est que l'approche par modèle dans DARE implique toujours une démarche d'abstraction. L'acteur doit accéder à la tâche, abstraire sa modification, la formaliser et l'appliquer. Ce processus peut s'avérer coûteux cognitivement, ce qui peut introduire un déséquilibre entre la motivation pour réaliser l'activité et l'effort à fournir pour adapter son contexte, ce qui correspond à un échec du système. Nous pensons qu'il serait bénéfique de mettre en place de nouveaux mécanismes qui facilitent la cristallisation d'expérience.

4.4 Le projet CoolDA

Le projet CoolDA (Bourguin, 2003)(Bourguin, 2004) consiste en une refonte totale de l'infrastructure DARE pour tenir compte tant des évolutions de nos modèles, suite à nos premières expériences, que des nouveaux contextes technologiques. Même si ces dimensions technologiques sont aussi importantes, notre objectif n'est pas d'exposer dans cet article toutes les spécificités de l'infrastructure CoolDA telle que développée aujourd'hui, en particulier son architecture logicielle, son mode de distribution, ses interfaces utilisateurs, etc. Nous avons choisi ici de mettre en exergue les choix plus conceptuels effectués en fonction des questions posées suite à nos travaux dans DARE. C'est pourquoi nous présenterons dans un premier temps de nouveaux modèles qui pallient le problème de décrochement conceptuel. Nous présenterons ensuite nos travaux sur les mécanismes de son nouveau noyau réflexif.

Méta modèle et modèle générique de Support d'Activité

CoolDA doit supporter différents types d'activités et donc différents types de SA peuvent être créés. Dans ce dessein, le système repose sur un modèle objet générique de SA qui peut être spécialisé pour des besoins spécifiques. On y retrouve les concepts décrits dans le modèle conceptuel présenté précédemment (cf. Figure 4). Notre objectif est de supporter la co-évolution du SA pendant son exécution. Dans une conception traditionnelle, le modèle générique ne serait connu que par les concepteurs du système. De ce point de vue, ce modèle est un ensemble de classes abstraites qui peuvent être spécialisées grâce à un langage de programmation : les notions de classe, d'attribut ou encore de méthode doivent être maîtrisées. La question est : « comment rendre notre modèle générique spécialisable du point de vue des utilisateurs finaux ? ». Ceci est réalisé grâce au méta modèle de CoolDA.

Un méta modèle est un modèle de modèle. Par exemple, UML définit un méta modèle décrivant les entités et relations pour construire des modèles objet. Le rôle de notre méta modèle est similaire à celui de UML. La différence est qu'il définit les bases d'un langage orienté vers notre domaine d'intérêt, c'est-à-dire la modélisation de SA. Nous avons montré qu'un SA est l'instance d'une tâche. En d'autres termes, le modèle d'un SA particulier est une tâche particulière. C'est pourquoi notre méta modèle réifie ce qu'est une tâche, c'est-à-dire quelle est sa structure et quels sont ses composants. Ce modèle⁴ est représenté dans la Figure 9.

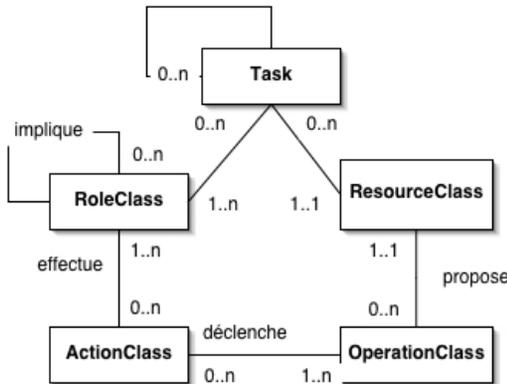


Figure 9. Le modèle de tâche.

La tâche est liée à un ensemble d'autres tâches (qui peut être vide), une ressource et un ensemble de rôles. Un rôle dans une tâche peut impliquer un autre rôle dans une autre tâche pour celui qui le joue. Chaque rôle pourra en fonction de l'état de l'activité effectuer des actions sur la ressource de sa tâche. Chaque action est décomposée en opérations proposées par la ressource. Du point de vue de la réalisation, notre hypothèse est que dans CoolDA, et dans un premier temps, les ressources représentent des outils disponibles sous forme de composants Java disponibles sur l'Internet au travers de leur URL. De plus amples informations sur ce

⁴ Ce modèle conserve une notation en langue anglaise car les entités nommées ici représentent des objets du système à ne pas confondre avec les objets monde réel qu'ils représentent. De plus, nous nous rapprochons ainsi des conventions de nommage utilisées en génie logiciel, en particulier celle du langage Java.

type de composants peuvent être trouvées dans (Bourguin, 2004). Les opérations permettent de déclencher dynamiquement des méthodes du composant tout en fournissant un niveau d'abstraction plus élevé que celui du langage Java : une opération possède un nom non contraint par la grammaire du langage sous-jacent, une description, etc. La majuscule au début de chaque terme indique qu'il s'agit d'entités qui peuvent être instanciées, ce modèle étant un méta modèle. Ainsi, une tâche *cours* sera instance de l'entité *Task* qui décrit le modèle de toutes les tâches. Le suffixe *Class* indique que les instances des entités de ce méta modèle sont des modèles. Par exemple *Enseignant*, instance de *roleClass* décrit une classe de rôles. Une instance de *Enseignant* sera jouée par *Greg* dans son cours. Ainsi, l'entité *Task*, pourrait aussi être dénommée *ActivityClass*. Grâce à ce méta modèle, la modification d'un SA peut être réalisée non pas dans les termes d'un langage objet, mais en termes dépendants du domaine, *i.e.* tâche, outil, rôle, etc.

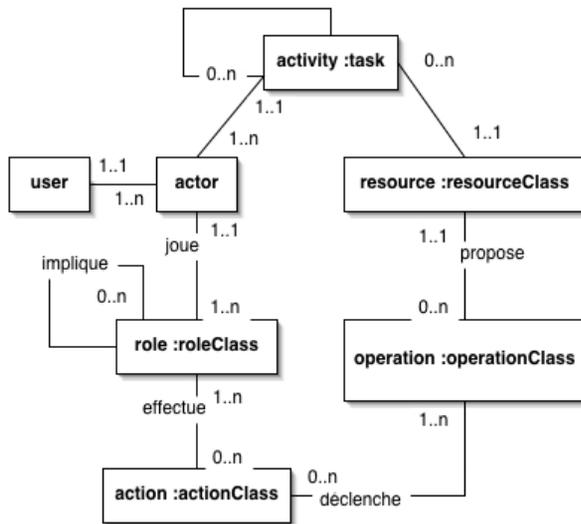


Figure 10. Le modèle d'activité.

La Figure 10 décrit le modèle générique de SA. Une activité est instance d'une tâche elle-même instance de l'entité *Task*. Un acteur de l'activité est l'association d'un utilisateur du système et de rôles définis dans la tâche. Un utilisateur peut être impliqué dans diverses activités du système. Chaque activité peut être liée à d'autres activités, ce qui signifie que certains de ses acteurs jouent un rôle qui les implique dans une autre activité. Les Figure 9 et 10 forment un noyau qui définit l'ensemble des entités du noyau réflexif. Il faut souligner que ce modèle ne répond pas au problème de séquençement des activités de type Workflow. Notre premier objectif est de permettre la description d'organisations et de fournir un environnement qui contextualise les ressources des utilisateurs en fonction de leurs rôles dans leurs activités.

Une particularité majeure de ce modèle est que chaque tâche/activité ne possède qu'une unique ressource. Ce point ne pose pas de contradiction avec nos inspirations de la TA car cette dernière ne place pas de limite rigide entre les niveaux action et activité. Nous abordons d'ailleurs le composant logiciel pointé par la ressource *CooLDA* lui-même comme un support d'activité prédéfini. En effet, un composant de mail supporte d'une manière autonome une activité de mail. Il définit

implicitement, au moyen de son interface et de ses fonctionnalités, le rôle de l'utilisateur dans l'activité qu'il supporte. Le rôle de CooLDA n'est autre que d'intégrer cette activité liée à une ressource particulière dans des activités plus larges. Par exemple, un *enseignant* utilise le composant de mail pour proposer une date d'examen à ses *étudiants* dans son activité *cours*. Cette approche nous permet de lever le problème de décrochement conceptuel exposé en 4.3. La structure de base est l'activité liée à l'outil par le biais de la notion de ressource. Une méta activité, impliquant un outil méta, est alors une activité à part entière qui peut donc intégrer des (*sous*) méta activités et s'articuler avec son activité de base pour supporter la co-évolution. La Figure 11 expose ce principe d'articulation inter-activités. Une communauté A utilise le support d'activité proposé par CooLDA pour réaliser son objet et produire l'artefact correspondant. Une communauté B a pour objet la production de la tâche qui définit l'activité A. Certains acteurs peuvent à la fois être membres des communautés A et B, mais ces deux communautés ne sont pas forcément les mêmes. Ainsi, l'activité A et sa méta activité B, sont intimement liées mais distinctes, ce qui fournit une gestion plus souple de la co-évolution par rapport à celle qui existait dans DARE.

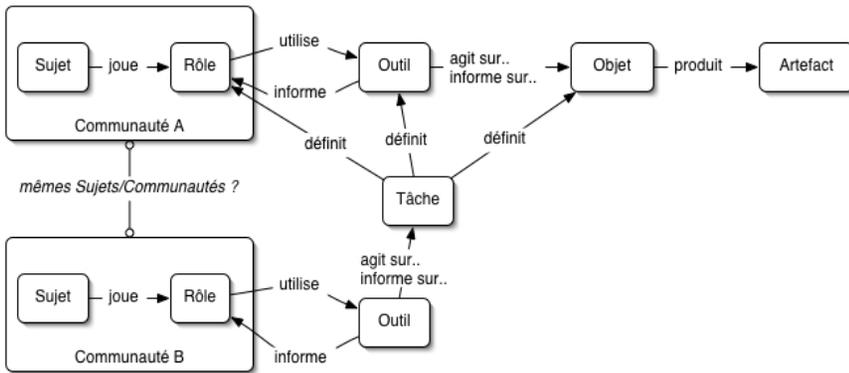


Figure 11. L'organisation liant une activité (communauté A) à sa méta activité (communauté B).

Des mécanismes relâchés

À partir des entités décrites dans notre modèle, nous devons créer un noyau réflexif qui propose une solution aux problèmes liés à la répercussion des actions effectuées dans une méta activité. Néanmoins, il nous faut aussi expérimenter des solutions au problème de l'approche par modèles et systèmes réflexifs que nous avons souligné précédemment (cf. 4.3). Nous avons donc implémenté totalement un nouveau noyau réflexif dans CooLDA. Ce noyau est inspiré des mécanismes classiques trouvés dans les langages à objets, mais s'en démarque par plusieurs aspects.

Chaque objet représentant une entité du modèle de tâche de CooLDA possède une interface permettant d'accéder en lecture ou en modification à son contenu. Par exemple, un modèle de ressource *Tableau Blanc* peut fournir la définition de l'opération *Démarrer*. Il est aussi par exemple possible de définir dynamiquement de nouvelles actions pour un type de rôle, ou encore de redéfinir les actions existantes. Chaque objet représentant une entité du modèle d'activité de CooLDA possède un état. Par exemple, une action sait si elle a déjà été déclenchée ou non. Chaque objet est de plus lié à un objet qui représente son modèle. Ainsi, une activité *cours de Greg*

connaît sa tâche *Cours*. Il en est de même pour les ressources, les rôles, etc. Par exemple, l'*enseignant* Greg peut accéder à sa définition, c'est-à-dire à un objet *Enseignant* de type *roleClass*. Ces mécanismes permettent l'introspection et l'intercession au cours de l'exécution du système : un support d'activité est capable d'accéder à son modèle d'exécution et de le transformer si le besoin s'en fait sentir.

Une méthode pour réaliser la relation causale entre les modèles et leurs instances est d'obliger chaque instance à consulter son modèle avant d'agir. Avec cette liaison dynamique, l'exécution de l'instance reste en concordance avec son modèle. Néanmoins, nous avons choisi d'assouplir cette approche et de parfois relâcher la relation causale. Dans le cas général, l'instance consulte effectivement son modèle. Toutefois, certaines propriétés décrites dans le modèle peuvent être surchargées, c'est à dire redéfinies dans l'instance. Ceci permet à une activité particulière, ou à un de ses composants, de se comporter comme un prototype qui, une fois stabilisé, peut être abstrait et cristallisé dans le modèle. Ce mécanisme est représenté dans la Figure 12 en l'appliquant à la modification d'un type de rôle.

Un exemple simple est celui de l'arrangement des fenêtres dans le support d'activité d'un acteur. L'acteur interagit grâce à des ressources dont la disponibilité est spécifiée dans son rôle, ou plus précisément, dans la classe de son rôle. L'acteur arrange visuellement ses ressources pour parfaire son espace de travail. Cet espace de travail résulte de ses préférences personnelles, mais aussi du fait qu'il joue un rôle particulier dans l'activité. Si l'acteur en possède le droit, il peut décider, par exemple après concertation avec sa communauté, de faire remonter son point de vue sur l'activité au niveau du modèle de son rôle. Cette action aura pour effet de rendre accessible son expérience aux autres utilisateurs qui jouent le même rôle. CoolDA, en tant qu'artefact, aura cristallisé l'expérience de l'acteur.

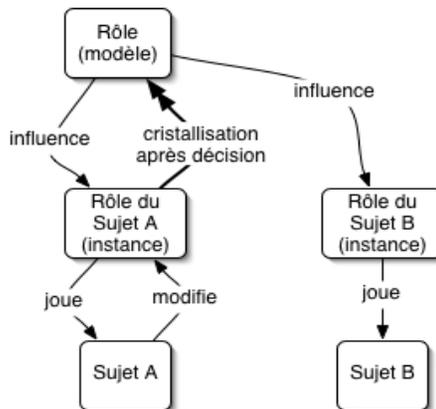


Figure 12. Evolution d'un rôle par abstraction d'un prototype.

Cet exemple expose l'effet bénéfique que peut avoir une relation causale relâchée. Grâce à ce nouveau mécanisme, notre approche permet toujours de cristalliser l'expérience des utilisateurs, mais l'environnement local peut être stabilisé avant qu'on puisse parler d'expérience partageable au travers du modèle. L'introduction de ce type de mécanismes est un élément de réponse au problème de stabilisation évoqué précédemment. C'est aussi un exemple de réponse au problème de l'effort d'abstraction : ici, l'expérience est cristallisée dans le modèle sans que l'utilisateur ait eu à passer par une description formelle de ses modifications.

La partie la moins avancée de nos réponses concerne la répercussion des modifications d'une tâche sur ses activités disjointes. Nous proposons un mécanisme de clonage qui permet de dissocier, à la demande des acteurs, leur tâche de celle dont elle est issue. Une tâche clonée ne possède plus de lien avec la tâche originelle et de ce fait, ses instances peuvent transformer leur modèle sans influencer le reste du système. Nous envisageons de coupler ce mécanisme avec un mécanisme d'héritage, non encore implémenté dans la version actuelle de CoolDA. Le clonage d'une tâche qui hérite d'autres tâches fournira une nouvelle tâche qui pourra évoluer de manière indépendante. La part et le choix de l'humain restent donc primordiaux dans notre approche. Ceci n'entre pas en contradiction avec nos objectifs : la co-évolution a pour but de supporter l'activité coopérative des utilisateurs, pas de tout automatiser. CoolDA n'est pas l'activité, il en fait partie.

4.5 Perspectives et comparaisons avec des travaux analogues

Quelques pistes de travail à court terme

Comme nous l'avons souligné dans l'introduction, CoolDA doit nous permettre prochainement de mener des expérimentations sur le terrain avec des utilisateurs en situation réelle dans le cadre des programmes de recherche coopératifs régionaux. Dans ce but, cette infrastructure est encore en cours de développement et nous travaillons à l'introduction d'une gestion des artefacts produits dans les activités ainsi que sur les moyens qui permettront, à la demande des acteurs, d'en gérer la dimension temporelle. De plus, l'utilisation de notre infrastructure suppose l'intégration dynamique de collecticiels. Nous avons déjà développé des composants intégrables dans une approche JavaBeans (Bourguin, 2004) tels une messagerie instantanée, un tableau blanc partagé ou encore un logiciel de partage pair à pair. Cependant, de manière à étendre la capacité d'intégration de CoolDA à des collecticiels tiers, nous nous inspirons aujourd'hui des travaux autour des Services Web pour développer de nouveaux composants. Nous ne savons pas si cette technologie sera la réponse au problème de l'intégration dynamique d'activités supportées par les collecticiels dans des systèmes de gestion d'activités plus globales, mais les propositions qui tentent de fournir aux utilisateurs les moyens de conjuguer eux-mêmes divers services disponibles sur le Web (Arsanjani et al., 2002) sont porteurs d'espoir dans le cadre de nos travaux.

Comparaison avec des travaux analogues

L'utilisation des mécanismes de la réflexivité pour supporter l'évolution des infrastructures informatiques n'est pas nouvelle en soi et a déjà fait l'objet d'intérêt notamment dans la communauté en Intelligence Artificielle. Dans nos champs d'applications privilégiés, le support aux Pratiques Collectives Distribuées (PCD), cela converge avec d'une part les approches de type Technomethodology⁵ proposé par Dourish et Button (1998) et, d'autre part, le point de vue de la recherche sur les pratiques de développement réflexif des systèmes d'information (Mathiassen, 1998). Dans le cas des travaux de Dourish, cette nécessaire réflexivité de l'action humaine située prend ses fondements dans l'ethnométhodologie. Il veut, pour cette science issue de la sociologie, comme nous pour la Théorie de l'Activité, dépasser les applications en tant que cadre d'analyse ou de critique des usages pour en faire un principe de conception des systèmes au travers de la réification de concepts clés de l'ethnométhodologie dans la technologie. Nous avons depuis longtemps étudié avec intérêt les travaux de Dourish tant ils présentaient de similitudes avec les nôtres, dans

⁵ Approche tentant de fonder l'informatique sur des apports de l'Ethnométhodologie.

leurs finalités et la démarche de conception, et aussi de différences quant aux cadres conceptuels et technologiques mobilisés. Une comparaison détaillée de ces deux approches nous semble hors de propos ici. Il convient de signaler que d'autres champs théoriques issus des sciences humaines, en particulier nord-américaines, ont également été mobilisés comme fondateurs pour la conception d'infrastructures ouvertes, évolutives pour le TCAO. Nous voulons plus particulièrement mentionner les travaux de Fitzpatrick et Kaplan qui prennent leurs fondements aussi dans les sciences sociales notamment dans le courant appelé « interactionnisme social/symbolique » (Fitzpatrick, et al., 1995) (Kaplan et al., 1998) (Kaplan, Seebeck, 2001).

5 Conclusion

Il est clair que notre travail de recherche, portant sur des infrastructures informatiques capables de supporter la co-évolution au sein des systèmes interactifs, est ambitieux et n'a pas encore fait la preuve de sa pertinence sur le terrain. Néanmoins, il est possible de dégager des résultats ou des questionnements utiles à la communauté de recherche sur les IHM. Nous pouvons établir ce bilan provisoire, obtenu à partir de la spécificité de notre démarche, selon trois niveaux d'analyse allant du particulier, la conception et la réalisation des infrastructures DARE et CoolDA, au plus général, celui d'une contribution à une autre vision, complémentaire de celles qui existent déjà en matière de conception de techniques plus anthropo-centrées, ceci en passant par la conception propre aux IHM.

5.1 Les leçons tirées de la conception et de la réalisation des projets DARE et CoolDA

Une des principales leçons est la compréhension que toute activité coopérative est intimement liée à une méta activité coopérative. Même si DARE se démarquait des autres réalisations en intégrant cette méta activité au sein des activités qu'il supporte, l'erreur que nous avons commise est de ne pas avoir complètement considéré cette méta activité comme une activité à part entière. Grâce à la définition d'un nouveau modèle conceptuel minimal récursif, CoolDA résout ce problème et envisage les activités comme une organisation d'activités plus ou moins fortement connectées dans un réseau dynamique non forcément hiérarchique.

L'autre grande leçon est que l'approche par modèles pour supporter les propriétés réflexives et de cristallisation de l'expérience des activités humaines semble bien appropriée à l'objectif de co-évolution. Cependant, notre culture informatique a influencé notre vision des mécanismes à mettre en œuvre. Les mécanismes d'introspection, d'intercession et d'héritage paraissent forts intéressants au niveau du développement des supports d'activité, mais il s'avère bénéfique d'en envisager des variantes ou versions relâchées qui supportent mieux les besoins des infrastructures supportant la co-évolution. CoolDA pourrait d'ailleurs s'inscrire dans l'approche Model Driven Architecture ou MDA fortement mise en avant par l'Object Management Group (OMG, www.omg.org) et il sera intéressant à l'avenir de voir dans quelle mesure notre approche par modèles et méta modèles pour supporter les activités coopératives humaines se démarque des travaux actuels du génie logiciel.

5.2 Contribution à la science des IHM

Notre recherche est une contribution à la science des IHM dans la mesure où elle attire l'attention tant sur les dimensions sociales et contextuelles de l'interaction avec des ordinateurs, que sur le caractère toujours plus expansif, dynamique, évolutif des besoins et outils mobilisés dans des pratiques toujours collectives. Bien

évidemment, notre domaine d'investigation est très tourné vers le TCAO où cette dimension collective est inhérente à l'activité humaine considérée. Mais nous affirmons qu'il n'y a pas d'outil ou d'interface « mono utilisateur » si l'on veut bien considérer : qu'il y a de moins en moins de barrières entre conception/concepteurs et usage/utilisateurs (les usagers achèvent, façonnent la conception dans l'usage) ; et que ces outils informatiques sont tous les jours de plus en plus dépendants d'infrastructures informatiques distribuées, dont l'Internet est l'archétype, et dont les contraintes technologiques, la gestion, la maintenance demandent des efforts collectifs (Sanduski, 2003). Cela revient pour nous en particulier à revoir les définitions et les rapports dialectiques qu'entretiennent Tâche et Activité, deux concepts au cœur des IHM depuis l'origine.

La TA s'avère très intéressante pour fonder notre re-conception des systèmes interactifs évolutifs car elle place la médiation par les instruments (techniques ou psychologiques) au centre de l'activité. Nous sommes conscients de ne pas avoir tiré parti de tous les apports de la TA, que ce soit aux sources, comme dans les travaux de Vygotski, notamment *Pensée & Langage* (Vygotski, 1997) et de Leontiev, ou que ce soit dans des dérivations intéressantes comme la théorie instrumentale (Rabardel, 1995) (Folcher, 2003) (Rabardel, 2001) (Beguïn, 2003). Comme Rabardel en fait la demande, il faudrait développer une théorie instrumentale généralisée dont l'objectif serait « de rassembler et d'organiser en un ensemble cohérent (mais pas nécessairement non contradictoire) ce que nous savons aujourd'hui de l'activité humaine considérée sous l'angle des moyens, de quelque nature qu'ils soient, c'est-à-dire des instruments que les sujets s'approprient, élaborent et mobilisent au sein de l'activité, des actions et opérations en tant que médias de leurs réalisations » (Rabardel, 2002, page 272). Dans ce programme de recherche pluridisciplinaire, les spécialistes des IHM ne doivent pas seulement attendre des résultats. Ils doivent aussi y contribuer par leur expérience dans la modélisation et la réalisation des IHM. Il nous apparaît que le concept d'instrument, déjà fortement théorisé, vu comme un « biface » outil/signe, n'a pas encore livré toute sa complexité, toute sa richesse, utiles à la conception des IHM et à la compréhension des logiques d'interaction. Cette approche instrumentale est voisine de celle dite de l'*interaction instrumentale*, obtenue par des voies différentes, et proposée comme un modèle visant à dépasser les interfaces WIMP (Windows, Icons, Menus and Pointing device) (Beaudouin-Lafon, 2000). C'est également dans cette direction, celle de la dimension sémiotique de l'instrument, que nous allons porter notre effort dans le développement conceptuel et technologique d'infrastructures comme CoolDA.

5.3 Une contribution à une refondation d'une conception plus démocratique de la technique

Nos travaux de recherche s'inscrivent, depuis de nombreuses années dans une vision nouvelle du développement des techniques, notamment des techniques informatiques. Il s'agit d'une adhésion à un programme de recherche, pris dans son sens politique. Dans le domaine de l'interaction entre humains et ordinateurs, cette volonté s'est exprimée dès 1962 dans les travaux d'Engelbart qui reconnaissait la puissance d'une approche co-évolutionnaire et proposait comme stratégie le *Bootstrapping* (Engelbart, 1962, 1988). Nous avons signalé au chapitre 2.2 que plusieurs chercheurs ont appelé un changement dans la vision de l'informatique. Le sous-titre du livre de Winograd et Flores (1986) n'était-il pas *New foundations for the design* ? On pourrait aussi mentionner le livre *Computer and Democracy* (Bjerknes et al., 1987). Devant la complexité des nouvelles technologies informatiques, de leur caractère de plus en plus intrusif, ubiquitaire, persuasif, il incombe à la communauté de recherche en IHM de maintenir parallèlement une démarche visant à réintégrer l'utilisateur non seulement en tant qu'acteur, mais aussi en tant que citoyen. Nous sommes conscients

que dans les grandes boucles évolutives présentées dans la Figure 1, c'est aujourd'hui la science et la société, qui doivent être en co-évolution (Nowotny et al., 2003). Cela appelle ce que Feenberg désigne en particulier sous le concept de technique réflexive, une vision plus démocratique de la technique où « *dans cet avenir, la technique ne sera pas un destin qu'il faut accepter ou rejeter, mais un défi à la créativité politique et sociale* » (Feenberg, 2004). C'est dans cette voie que modestement nous conduisons notre action de recherche.

6 Remerciements

Les auteurs tiennent à remercier tout particulièrement les organismes et les supports que ceux-ci leur ont apportés. En particulier la Direction de la Recherche pour l'Action Concertée Incitative Jeunes Chercheurs CoOLDev (Développement Coopératif du Logiciel), pour Grégory Bourguin, ainsi que le programme Technologies Avancées pour la Communication (TAC) financé par la Région Nord/Pas-de-Calais, le FEDER et l'Etat dans le cadre du Contrat de Plan Etat-Région (CPER), pour les deux laboratoires concernés.

7 Références

- Arsanjani, A., Chamberlain, D., Gisolfi, D., Konuru, R., Macnaught, J., Maes, S., Merrick, R., Mundel, D., Raman, T.V., Ramaswamy, S., Schaeck, T., Thompson, R., Diaz, A., Lucassen, J., Wiecha, C.F. (2002). (*WSXL*) *Web Service Experience Language Version 2*. IBM Note, 2002, <http://www106.ibm.com/developerworks/webservices/library/wswxl2/>.
- Bardram, J. E. (1997). Plans as Situated Action : An Activity Theory Approach to Workflow Systems. In *Proceedings of the Fifth European Conference on Computer Supported Cooperative Work*, Kluwer Academic Publishers, 1997, 17-32.
- Beaudouin-Lafon, M. (2000). Instrumental Interaction : An Interaction Model for Designing Post-WIMP user interfaces. In *Proceedings of CHI'2000 (The Hague, The Netherlands)*, ACM Press, 446-453.
- Bedny, G., Meister, D. (1997). *The Russian theory of activity, Current Applications to Design and Learning*. Lawrence Erlbaum Associates, Publishers.
- Béguin, P. (2003). Design as a mutual learning process between users and designers. *Interacting With Computer*, vol. 15, 709-730.
- Bertelsen, O. W. (1998). Elements to a theory of design artefacts: a contribution to critical systems development research. Ph.D. Thesis, Aarhus University, DAIMI, 1998, PB-531.
- Bjerknes, G., Ehn, P., Kyng, M. (1987). *Computers and Democracy: A Scandinavian Challenge*. Aldershot, UK: Ablebury, 1987.
- Bodker, S. (1991). Activity Theory as a challenge to system design. In *Information systems Research: Contemporary Approaches and Emergent Traditions*, Nissen H., Klein H., Hirschheim R. (Eds.), Elsevier Science Publishers, BV (North Holland), 551-564.
- Bourguin, G. (2000). *Un support informatique à l'activité coopérative fondé sur la Théorie de l'Activité : le projet DARE*. Thèse de doctorat en Informatique, n° 2753, Université des Sciences et Technologies de Lille, France, 2000, 210 p.

- Bourguin, G., Derycke, A. (2000). A Reflective CSCL Environment with Foundations Based on the Activity Theory. In *Proceedings of ITS'2000, Fifth International Conference on Intelligent Tutoring Systems (Montreal, Canada, 19-23 June 2000)*, LNCS vol. 1839, Springer Verlag, 272-281.
- Bourguin, G., Derycke, A., Tarby, J.C. (2001). Beyond the Interface : Co-evolution inside Interactive Systems - A proposal founded on Activity Theory. In *Proceedings of IHM-HCI 2001 (Lille, France, 10-14 September 2001)*, Springer Verlag, 297-310.
- Bourguin, G. (2003). Les leçons d'une expérience dans la réalisation d'un collectifiel réflexif. In *Actes de la 15ème conférence francophone IHM 2003 (Caen, 24-28 Novembre 2003)*, ACM International Conference Proceedings Series, 40-47, ACM Press.
- Bourguin, G. (2004). Proposition pour une gestion dynamique de l'inter-activités dans le TCAO. In *Actes de IHM 2004 (30 août-3 septembre 2004, Namur, Belgique)*, ACM International Conference Proceedings Series, 191-194, ACM Press.
- Bowers, G., Leigh Star, S., Turner, W., Gasser, L. (1997). *Social science, technical systems and cooperative work: beyond the great divide*. Lawrence Erlbaum Associates, "Computer, cognition and work" series, 470 p.
- Carroll, J., Rosson, M. (1992). Getting around the Task-Artefact Framework: How to Make Claims and Design by scenario. *ACM Transaction On office Information Systems*, vol. 10 (2), 181-212.
- Christiansen, E. (1996). Tamed by a Rose : Computers as Tools in Human Activity, In (Nardi, 1996), 174-198.
- Clot, Y., (1999). *Avec Vigotski*. Edition la Dispute, Paris, 1999 (2002 deuxième version augmentée).
- Derycke, A. (1998). Integration of the learning processes into the Web: Learning Activity Centred Design and Architecture. In *Proceedings Webnet'98 conference*, invited conference, Orlando, FL, November 1998.
- Derycke, A., Kaye, A. (1993). Participative modelling and design of collaborative learning tools in the CO-LEARN project. In G. Davis, B. Samways (Eds.), *IFIP, Teleteaching 95 Conference (Trondheim, August 20-25)*, North-Holland, Amsterdam, 191-200.
- Dourish, P. (1998). Using Metalevel Techniques in a Flexible Toolkit for CSCW Applications. *ACM Transaction on Computer-Human Interaction*, vol. 5, n°2, 1998, 109-155.
- Dourish, P., Button, G. (1998). On "Technomethodology": foundational relationships between ethnomethodology and system design. *Human-Computer Interaction*, vol. 13, 395-432.
- Edward, K., Bellotti, V. Dey, A. K., Newman, M. (2003). Stuck in the middle: the challenges of user-centred design and evaluation of the infrastructure. In *Proceedings of CHI'2003 ACM conference (April, 5-10 2003, Ft Lauderdale, Florida)*, ACM Press, 297-304.
- Engelbart, D. (1962). *Augmenting Human Intellect : a Conceptual Framework*. Stanford Research Institute, 1962 (cité dans P&T Johnson-Lenz 1991).

- Engelbart, D. (1988). Bootstrapping and the Handbook Cycle. In *Proceedings of CSCW'88 (Portland, Oregon)*, ACM press.
- Engeström, Y. (1987). *Learning by expanding: an activity theoretical approach to developmental research*. Orienta Konsultit Oy, Helsinki.
- Feenberg, A. (2004). *(Re)penser la technique. Vers une technologie démocratique*. Edition la Découverte, Paris, février 2004.
- Ferraris, C., Brunier, P., Martel, C. (2002). Constructing collaborative pedagogical situations in classrooms : a scenario and role based approach. In *Proceedings Computer Support for Collaborative Learning 2002 (CSCL 2002)*, Boulder, Colorado, USA, 2002, 7-11.
- Fitzpatrick, G., Tolone, W. Kaplan, M. (1995). Work, locales and distributed social wOrlds. In H. Marmolin, Y. Sundblad, and K. Schmidt, editors, *Proceedings of the Fourth European Conference on Computer Supported Cooperative Work*, Kluwer Academic Publishers, 1-16.
- Folcher, V. (2003). Appropriating artifacts as instruments : when design-for-use meets design-in-use. *Interacting With Computer*, vol. 15, 647-663.
- Hollan, J., Hutchins, E., Kirsh, D. (2000). Distributed Cognition: Toward a New Foundation for Human-Computer Interaction Research. *ACM Transactions on Computer-Human interaction*, vol. 7, n°2, 174-196.
- Hughes, J., King, V., Rodden, T., Anderson, H. (1995). Ethnography in interactive systems design. *ACM Interaction*, April, vol. 11.2, 57-65.
- Hummes, J., Kohrs, A., Merialdo, B. (1998). Software Components for Cooperation: a Solution for the “Get help” Problem. In *Proceeding of COOP'98 conference*, INRIA, 1998.
- Johnson-Lenz, P., Johnson-Lenz, T. (1991). Post mechanistic groupware primitives: rhythms, boundaries and containers. In Greenberg (Ed.), *Groupware*, Academic Press, USA, 271-293.
- Kaplan, S., Fitzpatrick, G., Herring, Ch. (1998). Centers of collaborative work. In F. Darses and P. Zarate, (Eds.), *Proceedings COOP'98*, volume II, Cannes, France, 1998, INRIA, 89-92.
- Kaplan, S., Seebeck, L. (2001). Harnessing Complexity in CSCW. In *Proceedings of the seventh European conference on Computer Supported Cooperative Work (ESCW 2001, Bonn, Germany, 16-20 September 2001)*, Kluwer Academic Publishers, 359-378.
- Kaptelinin, V., Nardi, B., Macaulay, C. (1999). The Activity Checklist : a Tool for Representing the “Space” of Context. *Journal of Interactions*, July-August 1999, 27-39.
- Kaptelinin, V. (2003). UMEA: Translating Interaction Histories into Project Contexts. In *Proceedings of CHI 2003 (April 5–10, 2003, Ft. Lauderdale, Florida)*, CHI Volume No. 5, Issue No. 1, ACM Press, 353 - 360.
- Kiczales, G., Bobrow, D.G., Des Rivieres, J. (1991). *The Art of the Metaobject Protocol*. MIT Press, August 1991.
- Kirsh, D. (1999). Distributed Cognition, Coordination and Environment Design. In *Proceedings Of the European Cognitive Sciences Society* (<http://icl-server.ucsd.edu/~kirsh/articles/italy/published.html>).

- Kuutti, K. (1991a). The concept of activity as a basic unit of analysis for CSCW research. In *Proceeding of the second ECSCW'91 conference*, Kluwers Academics Publishers, 249-264.
- Kuutti, K. (1991b). Activity Theory and its applications to information systems research and development. In Nissen, H, Klein, H. Hirschheim, R. (Eds), *Information systems Research: Contemporary Approaches and Emergent Traditions*, Elsevier Science Publishers, BV (North Holland), 529-549.
- Kuutti, K. (1993). *Notes on systems supporting "Organisational context" – An activity theory viewpoint*. COMIC European project, deliverable D1.1, 101-117.
- Kuutti, K. (1996). Activity Theory as a Potential Framework for Human-Computer Interaction Research. In Nardi (1996), 17-44.
- Mathiasen, L. (1998). Reflective Systems development. *Scandinavian Journal of Information Systems*, vol. 10, n° 1 & 2, 67-117.
- Nardi, B.A. (Ed.) (1996). *Context and consciousness : activity theory and Human-Computer Interaction*. Cambridge, Ma : MIT Press.
- Nowotny, H., Scott, P., Gibbons, M. (2003). *Repenser la science*. Collection Débats, Belin, Paris.
- Rabardel, P. (1995). *Les hommes et les technologies : une approche cognitive des instruments contemporains*. Armand Colin, Paris.
- Rabardel, P. (2001). Instrument Mediated Activity in Situation. In *Proceedings of IHM-HCI 2001 (Lille, France, 10-14 September 2001)*, Springer Verlag, 17-30.
- Rabardel, P. (2002). Chapitre XIII « le langage comme instrument ? Eléments pour une théorie instrumentale étendue ». In Clot (version 2002), 265-289.
- Racithel, A., Velichkovsky, B. M. (1996). Joint Attention and Co-Construction: New Ways to Foster User-Designer Collaboration. In (Nardi, 1996), 199-233.
- Roussev, V., Prasun, D., Jain, V. (2000). Composable Collaboration Infrastructures Based on Programming Patterns. In *Proceedings of CSCW 2000*, ACM Press, 2000, 117-126.
- Saikali, K., David, B. (2001). Vers l'usage du Workflow pour la coordination dans les collecticiels. In *Proceedings of IHM-HCI 2001 (Lille, France, 10-14 September 2001)*, Springer Verlag, Springer Verlag, 399-411.
- Sanduski, R.J. (2003). Infrastructure Management as Cooperative Work : implications for System Design. *Computer Supported Cooperative Work*, vol. 12, 2003, 97-122.
- Shuurman, J.G., Heeren, H. *Participation and Connection in Collaborative Work*. Publication du Telematic Institute, Enchede, Nederland, GigaCSCW project, 20 p., <https://doc.telin.nl/dscgi/ds.py/Get/File-1255/Ppcneo2.pdf>.
- Senge, P.M. (1990). *The fifth Discipline: Art and Practices of the Learning Organization*. Currency Doubleday, New-York, 423 p.
- Sommerville, I., Bentley, R., Rodden, T., Sawyer, P. (1994). *Co-operative systems design*. The Computer Journal, vol. 37, N°5, 357-366.
- Suchman, L. (1987). *Plans and Situated Actions*. Cambridge University Press, Cambridge, UK.

Vicente, K. J. (2000). HCI in the Global Knowledge Based Economy: Designing to Support Worker Adaptation. *ACM Transactions on Computer-Human Interaction*, vol. 7, n°2, 263-280.

Vygotski, L. (1997). *Pensée & Langage*. Nouvelle traduction de F Sève, édition La Dispute, 1997, 537 p.

Wartofsky, M. (1979). *Representation and scientific understanding*. Dordrecht : Reidel.

Wild, P., Macredie, R. (2000). On Change and tasks. In McDonald S., Waern Y., Cockton G. (Eds.), *People and Computer XIV – Usability or Else, Proceedings of BCS HCI'2000*, Springer Verlag, 45-59.

Winograd, T., Flores, F. (1986). *Understanding Computer and Cognition*. Addison Wesley.