

Ingénierie des indicateurs d'activités à partir de traces modélisées pour un Environnement Informatique d'Apprentissage Humain.

Tarek DJOUAD*, Alain MILLE*, Christophe REFFAY**, Mohamed BENMOHAMED***

* LIRIS, Lyon, **LIFC, Besançon, ***LIRE, Algérie

■ **RÉSUMÉ** : Concevoir et exploiter des indicateurs d'activités constitue un enjeu fort pour l'apprentissage avec un EIAH. Nous proposons une approche de conception guidée par les modèles des traces d'interaction et exploitant un système de gestion de traces modélisées. Concevoir un indicateur consiste alors à modéliser explicitement la trace d'interaction permettant de le calculer, mettant en évidence les éléments d'observation à recueillir pour l'élaborer. Mettre en œuvre concrètement l'indicateur consiste à expliciter les transformations nécessaires à partir des traces d'interaction disponibles, considérées comme traces modélisées « premières » jusqu'à la formule de calcul de l'indicateur se fondant sur des observés de haut niveau. Un « chemin » de transformation est ainsi disponible et exploitable explicitement pour tout à la fois documenter et exécuter le calcul de l'indicateur. Une bibliothèque de modèles enrichis d'indicateurs (indépendants des plateformes) et des bibliothèques de transformations orientées indicateurs sont alors disponibles. Une illustration de l'approche sur la plateforme Moodle est proposée pour expliciter l'usage concret de la contribution proposée sur des indicateurs d'activité collaborative.

■ **MOTS CLÉS** : Trace modélisée, calcul des indicateurs, ingénierie dirigée par les modèles, apprentissage collaboratif.

■ **ABSTRACT** : Designing and exploiting collaborative learning activities indicators is a strong challenge for technology-enhanced learning. We propose an original approach, guided by the models of interaction's traces and exploited by Trace Based Management System. Designing an indicator consists in explicitly modelling the interaction's trace in order to provide observables for it. Therefore, effective implementation of the indicator consists in explaining the needed transformations on available interaction's traces (primary modelled traces), until being able to formulate the indicator computation. A «path» of transformations is explicitly designed and applied for both document and perform the calculation of the indicator. A library of enriched models indicators (platform independent) with their transformations sequence (context dependent) are available for reuse and share. Our approach is illustrated on the Moodle platform in order to explain the practical use of the proposed contribution for collaborative activities.

■ **KEYWORDS** : Modelled traces, indicator calculus, Model Driven Architecture, collaborative learning.

- [1. Introduction](#)
- [2. Calcul des indicateurs à partir des traces d'interaction](#)
- [3. Mise en œuvre](#)
- [4. Conclusions-Discussion](#)

1. Introduction

Évaluer les situations d'apprentissage collaboratif dans les environnements informatiques pour l'apprentissage humain (EIAH) est une tâche délicate qui nécessite une interprétation adaptée au contexte changeant de l'activité. Pour tenter de comprendre la dynamique de l'apprentissage, et évaluer efficacement ces situations d'apprentissage collaboratif, l'analyse des traces d'interaction est exploitée

par les chercheurs dans de nombreux systèmes dont un état de l'art avait été proposé dans (Soller et al., 2005). De la même façon, une modélisation adaptée des traces d'interactions devrait permettre aux enseignants de concevoir et d'automatiser le calcul des indicateurs, qui, à leur tour les aideront à comprendre, évaluer, suivre et soutenir l'apprentissage en cours.

Dans cet article, nous nous intéressons plus spécialement aux différentes méthodes et techniques facilitant l'analyse des traces dans les situations d'apprentissage en se basant sur l'ingénierie dirigée par les modèles (de traces), afin de concevoir et calculer efficacement les indicateurs indépendamment des plateformes d'apprentissage. Notre problématique de recherche comprend trois facettes : 1) comment récupérer et restructurer les traces brutes issues des sources de traçages pour donner naissance à de nouvelles traces modélisées nommées *traces premières* (Mille et Prié, 2006), c'est une phase de collecte des données permettant à travers la trace première, d'identifier et de sélectionner les données issues de l'EIAH, nécessaires au calcul de l'indicateur ; 2) quelles sont les transformations nécessaires au sens de l'Ingénierie Dirigée par les Modèles (IDM), et quels sont les opérateurs à définir, pour calculer, à partir de cette trace première, les indicateurs qui fassent sens dans le cadre d'une activité d'apprentissage; 3) quels sont les modèles de trace à associer aux indicateurs d'apprentissage recueillis dans la littérature EIAH et tout particulièrement aux indicateurs de collaboration, quelles données doivent-ils contenir et comment celles-ci doivent-elles être structurées pour permettre le calcul de l'indicateur visé. Ces transformations et ces modèles d'indicateurs peuvent être utilisés par le chercheur, le concepteur de l'activité, le formateur ou même l'apprenant en situation d'apprentissage.

Nous nous intéressons plus particulièrement à l'application de notre approche à la conception d'outils de prise de conscience de la collaboration (*collaboration awareness*) pour les différents acteurs en se basant sur le calcul d'indicateurs de collaboration en EIAH. Nous avons choisi cette application comme valorisante pour notre approche car illustrative de situations nécessitant une interprétation des dynamiques d'action collaborative plutôt qu'une interprétation d'un niveau d'apprentissage par exemple. Ce travail reprend les principes de l'ingénierie dirigée par les modèles pour transformer les modèles de trace et construire ainsi les indicateurs de la collaboration.

1.1. Les indicateurs dans les EIAH

Selon (Dimitracopoulou et Bruillard, 2006) un indicateur est une variable au sens mathématique à laquelle est attribuée une série de caractéristiques. C'est une **variable** qui prend des valeurs de **forme** numérique, alphanumérique ou même graphique.... La valeur possède un **statut** : elle peut être brute (sans unité définie), calibrée ou interprétée. Le statut identifie une caractéristique bien précise : celle du **type d'assistance** offert aux utilisateurs. Chaque indicateur peut dépendre d'autres variables comme le temps, ou même d'autres indicateurs.

Dimitracopoulou dans (Dimitracopoulou, 2004) propose de calculer les indicateurs par des outils nommés outils d'analyse. La figure 1 explique le principe de fonctionnement de ces outils d'analyse. Les utilisateurs (formateurs, apprenants, etc.) des environnements d'apprentissage utilisent les différents modules d'activités ou de communication (e.g. : tableau blanc, chat, forums, etc.) La sélection des données permet de récupérer d'une façon automatique ou semi-automatique les traces de l'utilisation des modules d'activités par les différents utilisateurs. On peut récupérer ces données à partir, par exemple, des fichiers logs de la plateforme d'apprentissage étudiée. Le choix des données à sélectionner dépend des entrées de la méthode d'analyse qui calcule l'indicateur.

Les méthodes d'analyse produisent un ou plusieurs indicateurs. L'indicateur peut indiquer le mode ou la qualité de la contribution individuelle (e.g. : Envoyer un message dans un Chat), de la collaboration (e.g. : la répartition du travail, la densité ou la cohésion d'un groupe, etc.) ou encore le processus ou la *qualité* du produit final (e.g. : la profondeur d'un fil de discussion dans un forum, etc.).

La valeur de l'indicateur permet de construire un retour (*feedback*) plus ou moins élaboré aux utilisateurs. Selon les catégories proposées par (Soller et al., 2005), ce retour peut être une simple visualisation de la valeur de l'indicateur (*mirroring*), ou cette valeur peut être comparée avec une valeur souhaitée (*monitoring*) ou encore servir à la construction d'une réponse plus élaborée pour guider l'apprenant dans son apprentissage (*guiding*).

De nombreux travaux ont été publiés sur les indicateurs respectant en général la définition proposée par (Dimitracopoulou, 2004). Citons par exemple (Reffay et Lancieri, 2006) pour le calcul de la cohésion et

la centralité dans les réseaux sociaux et à partir des forums de discussion. La plateforme ACOLAD (Jaillet, 2005) fournit au tuteur un outil le renseignant sur le triplet d'activité (Assiduité, Disponibilité, Implication). (Santos et al., 2003) proposent un outil qui calcule à partir des interactions, le degré d'implication de chaque apprenant dans la formation. Il identifie : l'apprenant participatif, perspicace, utile, non-collaboratif, qui prend des initiatives, et l'apprenant communicatif. D'autres indicateurs sont interprétés qualitativement comme (Martinez et al., 2003) où l'indicateur de la densité du réseau social est interprété à l'aide d'histogrammes. Dans (Tedesco, 2003), on calcule l'accord et le désaccord entre les apprenants.

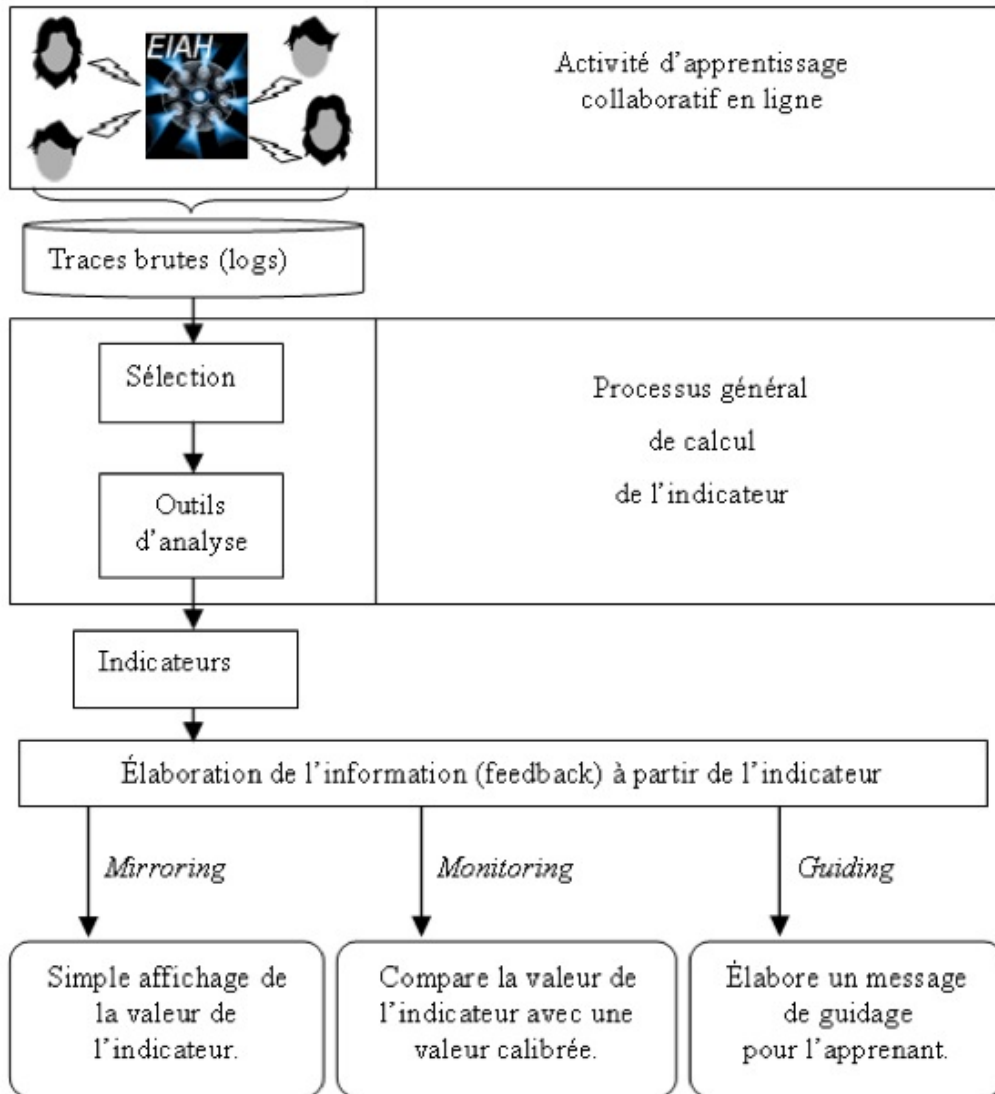


Figure 1 • Positionnement du calcul d'un indicateur

(May, 2008) propose une visualisation de l'indicateur « lecture d'un message dans un forum » à partir des traces. Le diamètre de la sphère donne la mesure du temps passé par un utilisateur pour la lecture d'un message. La distance entre les sphères représente le temps écoulé entre deux lectures, et la couleur de la sphère donne une signification du type d'action faite par l'utilisateur (e.g. : la couleur bleue signifie que l'utilisateur a affiché le message, et qu'il a effectué une lecture partielle). Cette visualisation de l'indicateur permet une interprétation plus détaillée des différentes actions de lecture faites par les apprenants (Fig. 2).



Figure 2 • Visualisation de l'indicateur « Lecture d'un message » dans Travis

(Lavallard, 2008) propose une plateforme d'exploitation d'archives de forums « ForumExplor ». Cet instrument de visualisation et d'aide à la lecture d'archives des forums facilite l'analyse complexe des forums et propose différentes vues globales et thématiques d'archives de forums à la demande des utilisateurs. La vue globale est basée sur un calcul statistique simple. La figure 3 montre une vue globale d'un indicateur qui représente la participation en nombre de messages par jour des différents utilisateurs dans un forum.

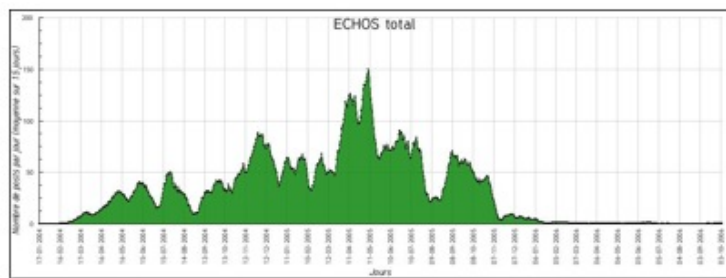


Figure 3 • Vue globale pour un indicateur de participation dans un forum

1.2. Travaux en relation

(Diagne, 2008) propose une architecture multi-agents pour réutiliser les indicateurs à partir de nouvelles sources de traçage. C'est une architecture ouverte structurée en agents : un agent requête qui interroge les sources de traçage et identifie les données collectées pour calculer la valeur de l'indicateur ; un agent base de données qui construit la trace des interactions ; un agent IHM pour afficher la valeur de l'indicateur. L'agent indicateur utilise la trace envoyée par l'agent base de données, il analyse cette trace pour calculer la valeur de l'indicateur (où l'indicateur est une fonction f), et passe par la suite les résultats à l'agent IHM pour afficher les résultats. La figure 4 explique le principe de cette architecture.

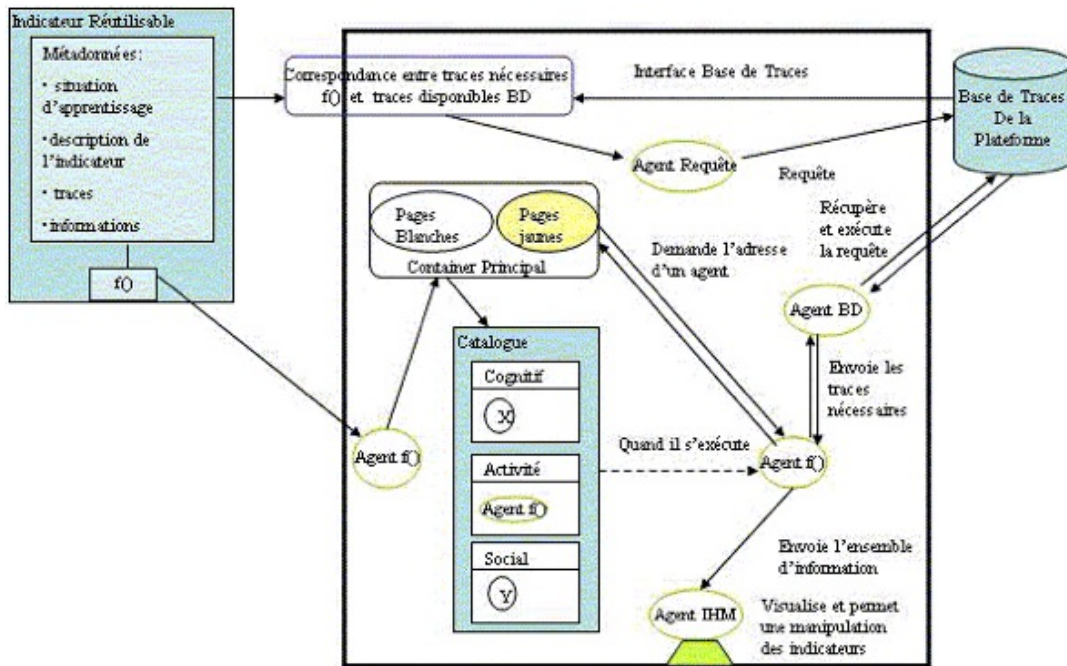


Figure 4 • Architecture du Système EM-AGIIR (Diagne, 2008)

Ce travail pose bien la question de la réutilisation d'indicateurs, mais considère que les traces disponibles pour le calcul d'indicateurs sont au bon niveau de formulation pour alimenter directement la formule de calcul de l'indicateur. Il n'est pas question de transformations des traces et encore moins de transformations explicites selon des modèles spécifiés.

UTL (*Usage Tracking Language*) est un langage qui permet « aux acteurs intervenant dans le cycle de vie d'un EIAH de décrire les traces d'usage et leur sémantique, ainsi que la définition des besoins d'observation et des moyens à mettre en œuvre pour l'acquisition des données à collecter » (Choquet et Iksal, 2007). UTL propose deux types de données : primaires et dérivées. Les données primaires sont directement collectées à partir de l'activité. Parmi les données primaires, les auteurs distinguent les données brutes (événements issus de l'environnement), les données de production (les productions des apprenants) et les données additionnelles (annotations par exemple). Les données dérivées, calculées à partir des données primaires (ou de données dérivées) sont distinguées selon qu'il s'agit de données intermédiaires à un calcul ou sont des indicateurs. Les outils d'analyse permettant de calculer les données dérivées, en particulier, les indicateurs à partir de données primaires et/ou d'autres données dérivées sont définis à partir de ce langage.

Ce travail pose implicitement la question des transformations en différenciant les données brutes et les données dérivées. Ces transformations ne font pas l'objet d'une sémantique particulière. Il s'agit d'une conception en boucle ouverte (toute modification des données est liée à une re-conception) n'intégrant pas la notion de système de gestion de traces modélisées assistant la tâche d'élaboration des transformations. Ce travail n'est pas spécifiquement associé à la conception d'indicateurs réutilisables mais fournit un cadre général pour une ingénierie guidée par les modèles pour la conception d'EIAH (et leur re-conception).

(Pham Thi Ngoc, 2008) utilise les indicateurs pour améliorer et réutiliser les scénarios pédagogiques. Pour y parvenir il propose une grammaire formelle pour calculer les indicateurs en reprenant les concepts proposés pour UTL. Le modèle de représentation de l'indicateur est basé sur trois facettes : *Defining* qui définit le besoin d'une observation, *Getting* qui définit les moyens de l'observation à mettre en œuvre pour l'acquisition des données et *Using* qui explique comment utiliser ces données. Il s'agit d'une approche génie logiciel pour définir les modèles qui seront utiles à la re-conception en facilitant la réutilisation. Il n'y a pas de notion de système de gestion de base de traces permettant d'exécuter les requêtes de transformation et de tracer les chemins de transformation facilitant leur réutilisation.

(Voisin et Vidal, 2007) proposent une architecture guidée par les modèles (Model Driven Architecture

MDA) pour la gestion des traces. Cette architecture est ouverte et basée sur des technologies web et permet de superviser les activités d'apprentissage ou toute autre activité qui exploite des objets pédagogiques. La supervision s'appuie sur une modélisation de l'EIAH qui est une extension du méta-modèle CIM (Common Information Model) ([DMTF, 1999](#)), et dont l'architecture est conforme au WBEM (Web Based Enterprise Management) ([WBEM, 1999](#)). Un EIAH sera donc composé de ressources, d'utilisateurs et de relations possibles entre les différentes entités. Les auteurs proposent aussi un modèle UML générique de traces pour les activités, basé sur les modèles CIM. Ce modèle permet de structurer et d'ajouter une sémantique claire aux données observées. L'architecture manipulant ce modèle facilite le partage et la réutilisation des traces collectées. L'architecture a été implémentée en utilisant les API de Moodle ([Moodle, 2008](#)) et INES ([INES, 2008](#)). La contribution porte sur une architecture modulaire, facilitant la mise en œuvre de traçage de différents modules. L'ingénierie des indicateurs reste à la charge des utilisateurs.

Reading Tutor ([Mostow et al., 2005](#)) permet à des enfants d'apprendre à lire à l'aide d'un tuteur intelligent qui utilise les lectures des élèves pour les corriger. Le tuteur humain peut par la suite exploiter les traces d'interaction stockées dans une base de données MySql avec un outil d'interrogation de requêtes SQL. Ce système intègre bien la notion de requêtes sur des traces pour obtenir des « indicateurs » de différentes natures. La construction de requêtes imbriquées (appelant des requêtes cataloguées par exemple) s'apparente à un processus de construction d'indicateurs avec des données brutes et des données intermédiaires. La capitalisation est possible sous la forme de requêtes cataloguées spécialisées pour tel ou tel indicateur. Toutefois, la réutilisation dans différents contextes est difficile du fait que les requêtes ne permettent pas de construire des traces transformées.

([Merceron et Yacef, 2004](#)) analysent les traces issues de logic-ITA. L'outil récupère les fichiers traces et les stocke dans une base de traces. Par la suite l'outil utilise des techniques de *data mining* et fournit différents indicateurs : les erreurs les plus fréquentes, les exercices validés par tous, les apprenants qui n'ont réussi à faire aucun des exercices entamés, les groupes d'apprenants par capacité, les apprenants qui ont réussi ou échoué regroupés selon leur profil (déduit par un chemin dans un arbre de décision), les erreurs qui s'associent souvent. Ces indicateurs peuvent aider les tuteurs à jouer un rôle pédagogique en proposant aux apprenants des activités adaptées à leur niveau, et un rôle cognitif en les aidant à corriger leurs *misconceptions*. Cette approche permet d'identifier les motifs fréquents porteurs de sens pour la construction d'un indicateur. Ils pourraient être considérés comme des transformations à réaliser pour obtenir tel ou tel indicateur à partir de traces brutes. Ces travaux pourraient alors être considérés comme une méthode d'ingénierie pour identifier des modèles de trace intéressants pour la construction d'indicateurs. Ils sont toutefois éloignés des préoccupations principales du travail présenté ici.

1.3. Les indicateurs dans les EIAH : quelques exemples typiques

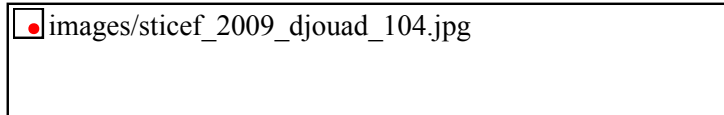
Nous détaillons dans cette section quelques indicateurs issus du domaine des EIAH. Tous ces indicateurs sont cités avec les plateformes où ils ont été implémentés dans ([Dimitracopoulou, 2004](#)). Ces indicateurs ont été choisis pour leur représentativité dans le cadre d'activités d'apprentissage collaboratif dont nous rappelons qu'il a été lui-même choisi car nécessitant d'élaborer des indicateurs d'apprentissage basés sur une analyse de l'activité (plus que sur l'analyse des productions). Ils sont choisis également pour la qualité de leur documentation qui permet d'en faire une re-description avec l'approche de transformations de traces que nous défendons. Nous avons privilégié les indicateurs effectivement implantés, raison pour laquelle on trouvera plusieurs fois les mêmes plateformes (en particulier ModellingSpace).

1.3.1. La division du travail (*Division of Labor*)

Cet indicateur défini et implémenté dans la plateforme COTRAS "*Collaborative TRAffic Simulator*" ([Jermann, 2004](#)) identifie la division du travail adoptée par deux utilisateurs qui agissent sur un ensemble de ressources. Cet indicateur destiné précisément aux chercheurs, permet d'identifier le rôle pris par chaque participant dans le processus de collaboration pour l'apprentissage. On peut distinguer trois types de division du travail : la division basée sur la *tâche* où chaque acteur agit séparément sur des ressources différentes. Une division basée sur le rôle, où un seul des deux acteurs agit sur toutes les ressources. Une division du *travail concourant* où les deux acteurs agissent (en même temps) sur l'ensemble des ressources.

La caractérisation de ces trois types de division du travail peut se calculer à partir de la somme des

différences (SD) et celle des différences absolues (SAD) :



Avec $S1A_i$ (respectivement $S2A_i$) : le total des actions faites par le sujet S1 (respectivement S2) sur la ressource A_i et $S1A$ (respectivement $S2A$) : le total des actions faites par le sujet S1 (respectivement S2) sur toutes les ressources. Le SAD indique la symétrie des actions. La valeur 0 signifie que les deux acteurs font le même nombre d'actions, alors que la valeur +1 signifie que toutes les actions sont faites par le même acteur, identifiable grâce au signe de SD. La figure 5 montre les différentes divisions du travail possibles calculées à partir des valeurs 0, 1 (et -1) des variables SD et SAD.

		Sum of Absolute Differences (SAD)	
Sum of Differences (SD)		0	1
1	N / A		Role
0	Concurrent		Task
-1	N / A		Role

Figure 5 • classification de la division du travail à partir des deux indicateurs SD et SAD. (Jermann, 2004)

1.3.2. Indicateur de repérage d'interaction

L'indicateur *Interaction* implémenté dans la plateforme ModellingSpace (Avouris et al., 2003) est utilisé par les enseignants pour mesurer le taux d'activité lors de la résolution d'un problème donné. Il calcule le nombre d'actions faites dans un module d'activité, dans un intervalle de temps. Si on considère l'intervalle $[t_0, t_m]$ associé à une session de collaboration, le temps est quantifié de la façon suivante : $t_i = t_0 + i * d$, avec $d = (t_m - t_0) / n$. Soit $Interactions(k, t_i)$ le nombre d'actions effectuées dans un module k durant l'intervalle de temps $[t_{i-1}, t_i]$ avec les valeurs de k $[1, k_{max}]$ correspondant aux outils d'interaction $\{k=1 \Rightarrow \text{chat}, k=2 \Rightarrow \text{forum}, \dots\}$.

Si $Interactions(k, t_i) = 0$ alors aucune action n'est faite sur le module d'activité k dans l'intervalle de temps $[t_{i-1}, t_i]$. Alors que si la valeur de $Interactions(k, t_i)$ est grande alors il y a *peut être* plus de collaboration entre les acteurs dans ce module.

1.3.3. Indicateur Agent Actif (Active Agent)

Cet indicateur implémenté dans la plateforme ModellingSpace (Avouris et al., 2003) est utilisé par les enseignants pour mesurer l'activité au cours de la résolution d'un problème donné. Un acteur n'est actif

que s'il interagit dans un module d'activité et dans un intervalle de temps. L'indicateur Agent actif, présente le *nombre* des acteurs qui ont interagi *au moins* une fois dans le module k sur un intervalle de temps donné. Nous pouvons calculer cet indicateur sur chaque module d'activité. Ainsi, $Agents(k, t_i)$ donne le nombre des acteurs qui ont interagi dans le module k , et dans l'intervalle de temps $[t_{i-1}, t_i]$.

1.3.4. Indicateur associé aux actions collaboratives (Collaborative Action function indicator CA)

Cet indicateur implémenté dans la plateforme ModellingSpace (Avouris et al., 2003) est utilisé pour représenter à la fois le nombre d'actions et le nombre d'agents actifs au cours de la résolution d'un problème donné. On calcule cet indicateur à partir de deux indicateurs précédents : $Agents(k, t_i)$ et $Interactions(k, t_i)$, sur un intervalle de temps t_i . L'indicateur CA est une somme sur l'ensemble des modules, dont les termes sont précisés par la formule suivante :

$$CA(t_i) = \sum_{k=1}^{k_{\max}} Agents(k, t_i) * Interactions(k, t_i)$$

1.3.5. Indicateur sur les actions non verbales (Non Verbal Actions NVA)

L'indicateur NVA implémenté dans la plateforme ModellingSpace (Avouris et al., 2003) est utilisé par les enseignants et les chercheurs durant et après la session de collaboration et représente le pourcentage de toutes les actions non verbales sur les différents outils d'interaction. L'idée est de calculer d'abord toutes les actions verbales (e.g. : message chat/forum, etc.), et ensuite de considérer le reste des actions comme actions non verbales. La formule de calcul suivante considère que les seules les actions verbales sont les actions de type chat. Mais nous pouvons élargir la règle pour le reste des actions verbales :

images/sticef_2009_djouad_108.jpg

Si $NVA \sim 0$ alors les acteurs ont tendance à n'utiliser que des actions verbales. $NVA \sim 1$ alors les acteurs ont tendance à n'utiliser aucune action verbale dans les outils dans l'intervalle de temps sélectionné.

1.3.6. Indicateur sur la contribution d'un acteur (Selected Agent Contribution SAC)

L'indicateur SAC implémenté toujours dans la plateforme ModellingSpace (Avouris et al., 2003) est utilisé par les enseignants, les chercheurs et les tuteurs pour évaluer la participation d'un acteur donné au cours de la résolution synchrone d'un problème posé. Il s'agit du taux d'interaction d'un acteur donné (Agent) par rapport à l'ensemble des interactions (de tous les acteurs) pour un module d'interaction k , et sur un intervalle de temps $[t_{i-1}, t_i]$. La formule de calcul est la suivante :

$$SAC(agent, t_i) = \begin{cases} \frac{\sum_{k=1}^{k_{\max}} Interactions(k, agent, t_i)}{\sum_{k=1}^{k_{\max}} Interactions(k, t_i)} & \text{si } \sum_{k=1}^{k_{\max}} Interactions(k, t_i) > 0 \\ 0 & \text{si } \sum_{k=1}^{k_{\max}} Interactions(k, t_i) = 0 \end{cases}$$

$Interactions(k, agent, t_i)$ étant le nombre des actions faites par un acteur visé (*agent*) dans le module k durant l'intervalle de temps $[t_{i-1}, t_i]$. SAC appartient à $[0, 1]$. Si $SAC = 0$ alors l'acteur n'a pas agi dans l'intervalle de temps considéré. Si $SAC=1$ alors l'acteur est l'unique intervenant dans l'intervalle de temps considéré.

1.3.7. Indicateur sur le pourcentage de participation (Participation percentage PART)

L'indicateur PART implémenté dans la plateforme ModellingSpace ([Avouris et al., 2003](#)) représente la proportion des agents ayant interagi dans au moins un module dans l'intervalle de temps considéré. La formule de calcul est la suivante :

$$PART(t_i) = \frac{Agents(t_i)}{TotalAgents}$$

Par exemple, si $PART(t_i)=0,5$ alors la moitié des agents ont interagi dans au moins l'un des modules sur l'intervalle de temps $[t_{i-1}, t_i]$. Si $PART(t_i)= 0$ alors aucun acteur du groupe n'a agi durant cette période.

1.3 Discussion et positionnement scientifique

Les travaux présentés dans la première partie de l'article proposent des contributions variées : proposition d'indicateurs pour les EIAH ([Reffay et Lancieri, 2006](#)), ([Martinez et al., 2003](#))... ; visualisation d'indicateurs et de traces ([May, 2008](#)) et ([Lavallard, 2008](#)) ; architectures et langages pour la réutilisation des indicateurs ([Diagne, 2008](#)), ([Voisin et Vidal, 2007](#)), ([Choquet et Iksal, 2007](#)), etc. Les indicateurs illustratifs présentés sont pour la plupart implantés dans la plateforme ModellingSpace ([Avouris et al., 2003](#)), et donnent des informations utiles aux chercheurs et aux enseignants. Aucun des travaux ne propose vraiment de méthodes génériques associant conception par les modèles et opérationnalisation de ces modèles par un module spécifique de type gestionnaire de traces modélisées.

Nous nous sommes fixé plusieurs objectifs dans cette recherche. L'un d'eux concerne la réutilisation des indicateurs dans des nouvelles plateformes et à partir des nouvelles sources de traçages, question également traitée dans ([Diagne, 2008](#)). Notre approche cependant très différente, est basée sur les transformations de M-Traces. Dans son travail, Diagne considère en revanche un indicateur comme le résultat d'une fonction de calcul sur les traces qui pourra être visualisé selon différentes techniques. La fonction de calcul encode complètement l'analyse des traces, ne permettant pas une réutilisation facile. Toujours dans ([Diagne, 2008](#)), une analogie entre Sélection-Analyse-Visualisation est faite avec le modèle d'objet « Modèle-Vue-Contrôleur », et on ne peut réutiliser un indicateur que dans des situations d'apprentissage similaires. Alors que notre approche est basée sur les transformations des M-Traces où nous associons pour chaque indicateur d'activité collaborative une M-Trace, et nous proposons une séquence de transformations qui fait le passage de la M-Trace première vers la M-Trace de l'indicateur visé. Nous pouvons par la suite réutiliser cette séquence de transformations pour recalculer le même indicateur dans d'autres plateformes d'apprentissage, ou même calculer un nouvel indicateur dans la même plateforme en réutilisant une partie de cette séquence.

Le calcul des indicateurs (présenté dans son contexte général sur la figure 1) nécessite un effort important. D'un côté, la phase de sélection des données nécessite une analyse fine des données dans les sources de traçage. De l'autre, un calcul spécial est associé à chaque indicateur, ce qui oblige à redéfinir ce calcul spécial pour chaque nouvel indicateur, mais aussi pour le même indicateur dans les différentes plateformes d'apprentissage, ce qui multiplie les efforts de construction et d'implémentation.

Pour alléger cette tâche, nous proposons une formalisation des indicateurs intégrant non seulement le mode de calcul, mais surtout le modèle des traces d'activité adapté à ce calcul. Nous présentons dans la section suivante notre méthode pour calculer les indicateurs par des transformations des modèles de trace d'interaction.

2. Calcul des indicateurs à partir des traces d'interaction

Nous présentons dans cette deuxième partie de l'article une méthode pour calculer les indicateurs avec un système à base de traces (SBT) ([Settoui et al., 2006](#)) et basée sur l'ingénierie dirigée par les modèles ([Nodenot, 2005](#)). Nous expliquons dans la première section, la démarche générale de notre méthode. Nous montrons dans la deuxième section en quoi elle s'appuie sur l'ingénierie dirigée par les modèles et en quoi elle nous est utile dans notre recherche. Nous finissons par présenter dans la troisième section, les trois différents cas possibles pour calculer un exemple d'indicateur avec notre méthode et en s'appuyant sur un SBT et sur l'ingénierie dirigée par les modèles de trace.

2.1. Démarche

Dans la méthode proposée ici, nous calculons un indicateur de collaboration à partir de transformations de modèles de trace. Cette méthode regroupe trois étapes : une collecte des données, des transformations de trace pour parvenir à la trace nécessaire au calcul de l'indicateur, et une étape de calcul. La figure 6 montre l'ordre de ces étapes :



Figure 6 • Démarche globale pour calculer un indicateur/trace

Avant d'expliquer chacune de ces trois étapes, nous présentons tout d'abord le principe d'un système à base de traces sur lequel s'appuie notre méthode.

2.1.1. Système à base de trace (SBT)

Un Système à Base de Traces ([Settoui et al., 2007](#)) est proposé par l'équipe SILEX pour gérer des traces modélisées. Une trace modélisée dans un SBT est décrite par un modèle de trace et un ensemble d'instances de ce modèle, où chaque instance est située sur l'axe du temps. On appelle *extension temporelle* associée à une trace : soit un intervalle temporel délimité par deux dates, appelées dates de début et de fin de l'observation ; soit une séquence d'éléments quelconques (par exemple une sous partie de l'ensemble des entiers naturels). Ainsi, on appelle *trace* une collection d'observés temporellement situés. On dénote par « *observé* » toute information structurée issue de l'observation d'une interaction. L'architecture du SBT ([Settoui et al., 2006](#)) regroupe : une base de traces modélisées (instances et modèle) ; le noyau du système offrant les outils de transformation de la trace ainsi que des modèles d'utilisation, le système de collecte construisant les traces modélisées à partir des sources de traçage, le système de visualisation des traces modélisées calculant les représentations à partir des traces modélisées selon les besoins des utilisateurs. Un gestionnaire des traces modélisées (M-Traces) ajoute et supprime des traces, définit les droits d'accès, et gère le graphe d'évolution des traces modélisées. Un système de requêtes permet d'interroger la base des M-Traces et de récupérer des informations adaptées aux besoins des utilisateurs. Une *transformation d'une trace modélisée* est tout processus qui transforme une M-Trace gérée par un système à base de M-Traces en une autre M-Trace gérée par le même système.

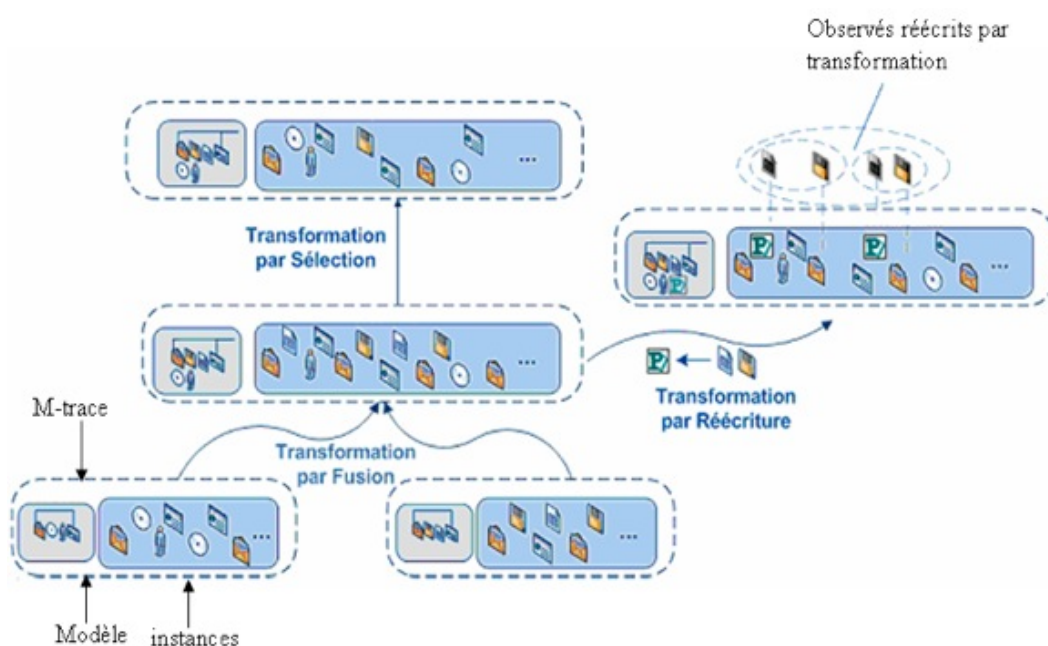


Figure 7 • Transformation¹ de M-Traces.

Les *M-Traces premières* d'une base de M-Traces d'un SBT sont les seules M-Traces non transformées de ce SBT. L'exploitation de la trace consiste en partie en sa transformation. La figure 7 illustre trois types de transformations de trace. Nous pouvons par exemple *fusionner* deux traces, le résultat est une trace qui regroupe les instances des deux anciennes traces. Nous pouvons par la suite *réécrire* le modèle de la trace résultat, faire des *sélections* sur les instances, etc. Nous détaillons plus loin dans cet article les opérateurs sur les M-Traces que nous proposons pour calculer les indicateurs.

2.2. L'ingénierie dirigée par les modèles

Un modèle dans (Seidwitz, 2003) est défini comme « *A set of statements about some system under study* ». Une autre définition dans (Bézivin et Gerbé, 2001) définit un modèle comme « *A simplification of a system built with an intended goal in mind. The model should be able to answer questions in place of the actual system* ».

Un modèle (Nodenot, 2005) est une *description* ou une *prescription* de tout ou partie d'un système à partir d'un langage clairement défini. Dans le cas de la description, le modèle qui décrit ce système est correct si ses caractéristiques et son comportement évoluent dans le temps de la même manière que le système réel. Alors que dans le cas de la prescription des systèmes à développer, le système est considéré comme valide si aucune caractéristique du modèle n'est contradictoire avec le système résultat. Et pour décrire les modèles, on utilise des méta-modèles. Un méta-modèle est le modèle qui définit le langage qui exprime le modèle visé (OMG, 2002).

L'ingénierie dirigée par les modèles en EIAH, inspirée du génie logiciel, a comme but de se focaliser sur les transformations apportées sur les modèles (selon leurs contextes d'utilisation), et non pas sur le code source du système, ce qui diminue largement l'effort des concepteurs, des enseignants, des chercheurs,... où tout le travail se capitalise sur les transformations des modèles qui prescrivent les systèmes et non pas sur les systèmes eux-mêmes. Outillées par des environnements de construction et de transformation de modèles, ces opérations peuvent être rendues accessibles aux utilisateurs du système à tracer, ce qui leur confère une bien plus grande flexibilité et adaptabilité aux besoins sans cesse renouvelés. Nous considérons que notre travail est directement lié à l'ingénierie dirigée par les modèles. En effet, la méthode de calcul de l'indicateur que nous proposons suppose d'une part la conception d'un modèle (de M-Trace) et d'autre part une séquence de transformations permettant de passer du modèle de la trace première au modèle de l'indicateur. Les différents opérateurs de comparaison et de transformation disponibles dans le SBT nous dispensent ainsi d'un passage fastidieux et *ad hoc* par les détails de codage. C'est ce qui constitue l'essentiel de la contribution de cet article dans le calcul des indicateurs par rapport aux méthodes actuelles.

2.3. Ingénierie des indicateurs dirigée par les modèles de trace

Avec les méthodes traditionnelles, les données sont préparées avant d'être traitées par un calcul spécial pour obtenir un indicateur. Ainsi, lors d'un changement (ou évolution) de la plateforme, ce même calcul spécial peut nécessiter un nouvel effort de codage. Nous sommes ainsi, devant la difficulté de recoder à chaque fois les étapes « Préparation des données » et « Calcul spécial » pour chaque nouvel indicateur, et ce, pour chaque plateforme d'apprentissage souhaitant utiliser cet indicateur.

Proposition

Notre proposition est d'associer à chaque indicateur « I » un modèle de trace permettant son calcul direct. Nous définissons un indicateur I par :

$$I = \{R_I, MT_{RI}\}$$

Avec : R_I : la règle de calcul, MT_{RI} : le modèle de la trace permettant le calcul.

La figure 8 illustre cette proposition, où nous associons une trace $Trace(I)$ à un indicateur de collaboration $Indicateur(I)$. Une trace $Trace(I)$ est constituée de son modèle de trace MT_{RI} et de son instance (séquence d'observés respectant le modèle). Le passage de la trace $Trace(I)$ vers la valeur de l'indicateur se fait à l'aide d'une règle de calcul R_I . Par exemple, on peut calculer l'indicateur « *Agent Actif* » à partir d'une

règle de calcul qu'on nomme $R_{AgentActif}$. La règle de calcul s'applique sur une trace respectant le modèle de trace $MT_{AgentActif}$ associé à cet indicateur.

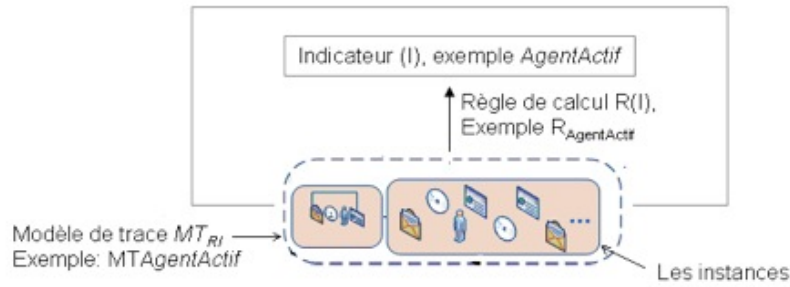


Figure 8 • Associer à chaque indicateur un modèle de trace décrivant les éléments nécessaires à son calcul

Nous proposons les étapes suivantes pour calculer un indicateur avec l'aide d'un SBT :

Etape 1 : Pour construire un nouvel indicateur « I » dans le SBT, nous proposons de définir un modèle de trace « I » ainsi que sa règle de calcul $R_I(x_i)$, paramétrée par l'ensemble des variables x_i , nécessaires au calcul de l'indicateur. Au départ, il n'existe aucune instance associée au modèle de l'indicateur, puisqu'il est en cours de définition. Dans cette étape (Fig. 9), nous associons à ce nouveau modèle une instance (vide à cette étape).

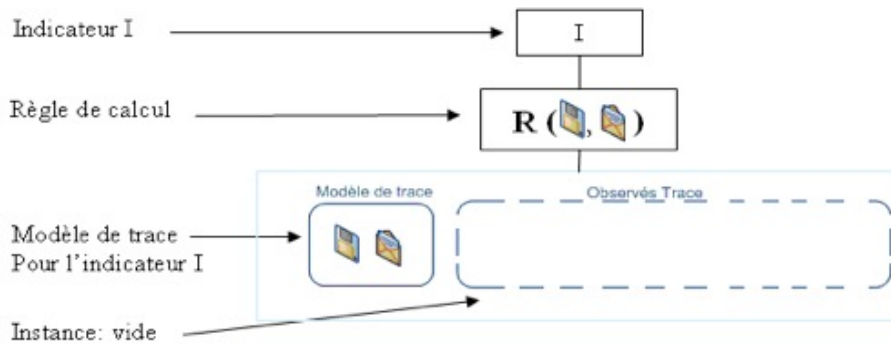


Figure 9 • Proposer un modèle de trace pour un nouvel indicateur « I »

Etape 2 : Dans une deuxième étape, l'utilisateur (ici, le concepteur de l'indicateur) est invité à définir une séquence de transformations pour l'indicateur « I » permettant de faire le passage de la trace première modélisée vers le modèle de l'indicateur « I ». Ce sont les opérateurs de transformation proposés par le SBT qui permettent de décrire cette séquence. La figure 10 illustre le principe de construction d'une séquence de transformations et montre ainsi la relation entre le modèle de la trace première et celui de l'indicateur. Des exemples concrets utilisant les opérateurs que nous proposons, sont présentés dans la section 3.

Etape 3 : La troisième étape nommée « préparation des données » permet de collecter les observés de la trace première conformément au modèle décrit dans l'étape 2. Cette trace dépend de la plateforme d'apprentissage et des différentes sources de traçage disponibles dans l'EIAH. La figure 11 illustre ce processus de collecte.

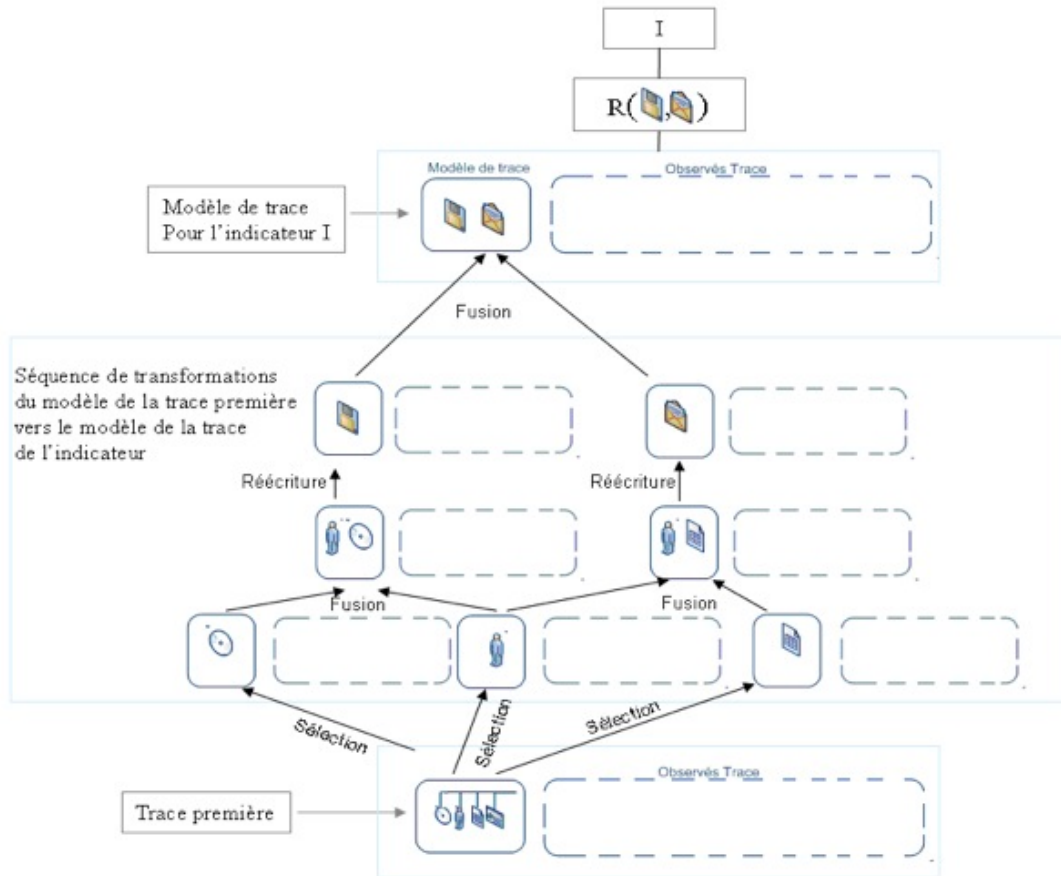


Figure 10 • Proposer une transformation de la trace première pour arriver au modèle de l'indicateur « I »

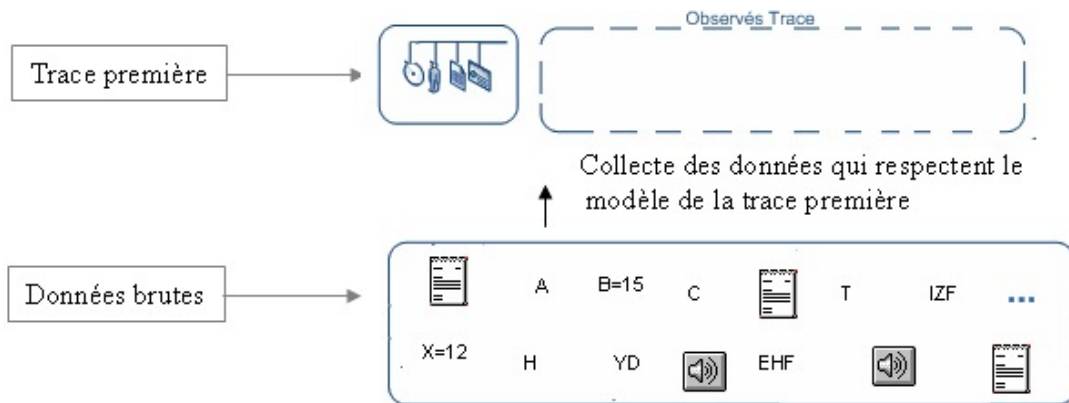


Figure 11 • Préparation des données :instancier le modèle de la trace première

Etape 4 : C'est la dernière étape, elle consiste à exécuter la séquence de transformations (définie dans l'étape 2) sur la trace première (instanciée à l'étape 3) dans le but d'instancier toutes les traces intermédiaires générées par la séquence de transformations. Le résultat de cette étape nous donne la trace de l'indicateur «I» conforme à son modèle de trace associé. La figure 12 illustre l'exécution de la séquence de transformations sur la trace première. Les observés de la trace de l'indicateur « I » ainsi obtenue servent de données d'entrée à la fonction « RI » (règle de calcul de l'indicateur) qu'il suffit d'appliquer pour obtenir la valeur de l'indicateur « I ».

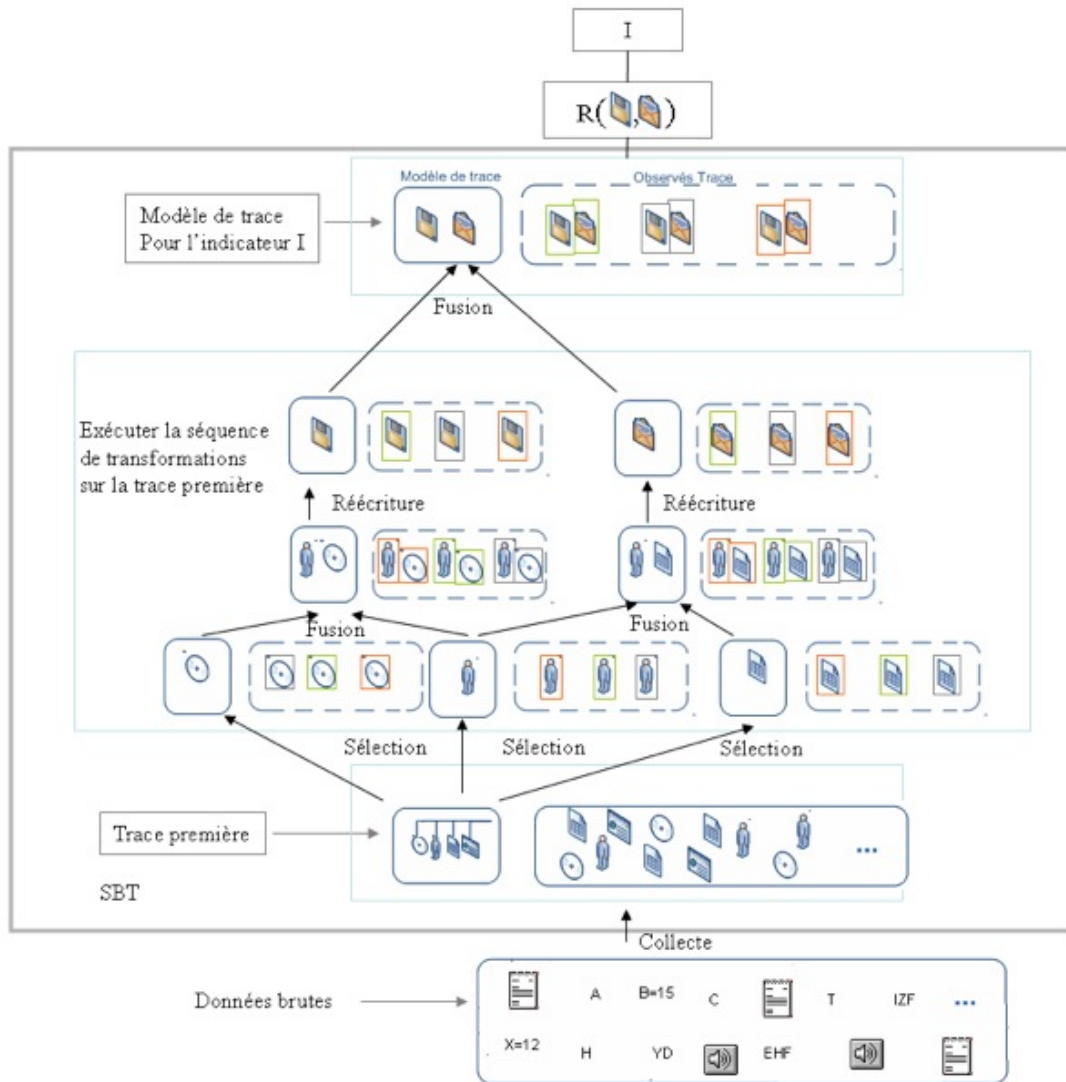


Figure 12 • Exécuter la séquence de transformations sur la trace première. Nous avons ici les quatre étapes regroupées : de la collecte au calcul de l'indicateur « I »

Nous remarquons aussi dans cette figure que les transformations, la trace première et la trace de l'indicateur sont toutes dans le SBT, alors que le calcul de la valeur de l'indicateur ainsi que les données brutes sont à l'extérieur du SBT. Le SBT devient comme une fonction qui a comme entrée les données brutes, et comme sortie la valeur de l'indicateur. À l'intérieur du SBT, il y a toutes les transformations (dirigées par les modèles de trace) pour faciliter la construction de l'indicateur visé.

Notre méthode, permettant des transformations sur les modèles, diminue largement l'effort pour calculer un indicateur. Les transformations des modèles minimisent cet écart entre les indicateurs et les sources de traçage, car nous nous focalisons sur les transformations des modèles, et non pas sur le calcul spécial associé pour chaque indicateur EIAH. La figure 13 compare le calcul d'un indicateur avec les méthodes *ad hoc* (que nous avons présentées précédemment), et le calcul d'un indicateur avec notre méthode. Nous pouvons remarquer que nous sommes sortis du cas traditionnel où les chercheurs passent par le calcul spécial spécifique à l'indicateur, vers toute une ingénierie basée sur les transformations des modèles de trace d'interactions.

	Collecter les données				Préparer les données				Calculer l'indicateur			
	Ad Hoc		SBT		Ad Hoc		SBT		AdHoc		SBT	
	Quoi	Qui	Quoi	Qui	Quoi	Qui	Quoi	Qui	Quoi	Qui	Quoi	Qui
Premier indicateur	Instru- menter	Info.	Instru- menter	Info	Calc Inter	Info	Trans Trace	Educ Info	Calc Final	Info	For- mule.	Educ
Nouvel Indicateur Même plateforme	Instru- menter	Info	-	-	Calc Inter.	Info	Réu- tiliser Trans.	Educ (Info)	Calc Final	Info	For- mule	Educ
Même indicateur Nouvelle plateforme	Instru- menter	Info	Instru- menter	Info	Calc Inter	Info	Réu- tiliser Trans.	Educ	Calc Final	Info	For- mule	Educ

Info : Informaticien - Calc : Calcul - Inter : Intermédiaire - Trans : Transformation - Educ : Educateur

Figure 13 • Comparaison entre le calcul ad-hoc d'un indicateur et notre méthode orientée transformation de modèles de trace.

Pour illustrer le bénéfice rendu par notre méthode s'appuyant sur l'ingénierie dirigée par les modèles, nous montrons comment créer un indicateur dans trois cas prototypiques : (1) créer entièrement un nouvel indicateur, (2) créer un deuxième indicateur en modifiant légèrement un indicateur existant et (3) réutiliser un indicateur existant pour l'adapter à une nouvelle plateforme.

2.3.1. Premier cas : Calcul d'un premier indicateur dans un SBT à partir d'une plateforme visée.

Il s'agit de calculer le premier indicateur dans le SBT, par exemple : l'indicateur « Agent Actif » sur une activité médiée par Moodle. Comme il s'agit d'un premier calcul, nous proposons de suivre la méthode composée des quatre étapes que nous venons de présenter : (1) proposer un modèle de trace pour cet indicateur, (2) définir la séquence de transformations associée, (3) organiser la collecte des données, et enfin (4) instancier des différents modèles issues de la transformation. La figure 14 montre un exemple complet de construction de l'indicateur « Agent Actif » sur la plateforme Moodle. Nous pouvons voir la souplesse de notre méthode et le passage des données brutes vers la valeur de l'indicateur, sans passer par un calcul spécial pour cet indicateur.

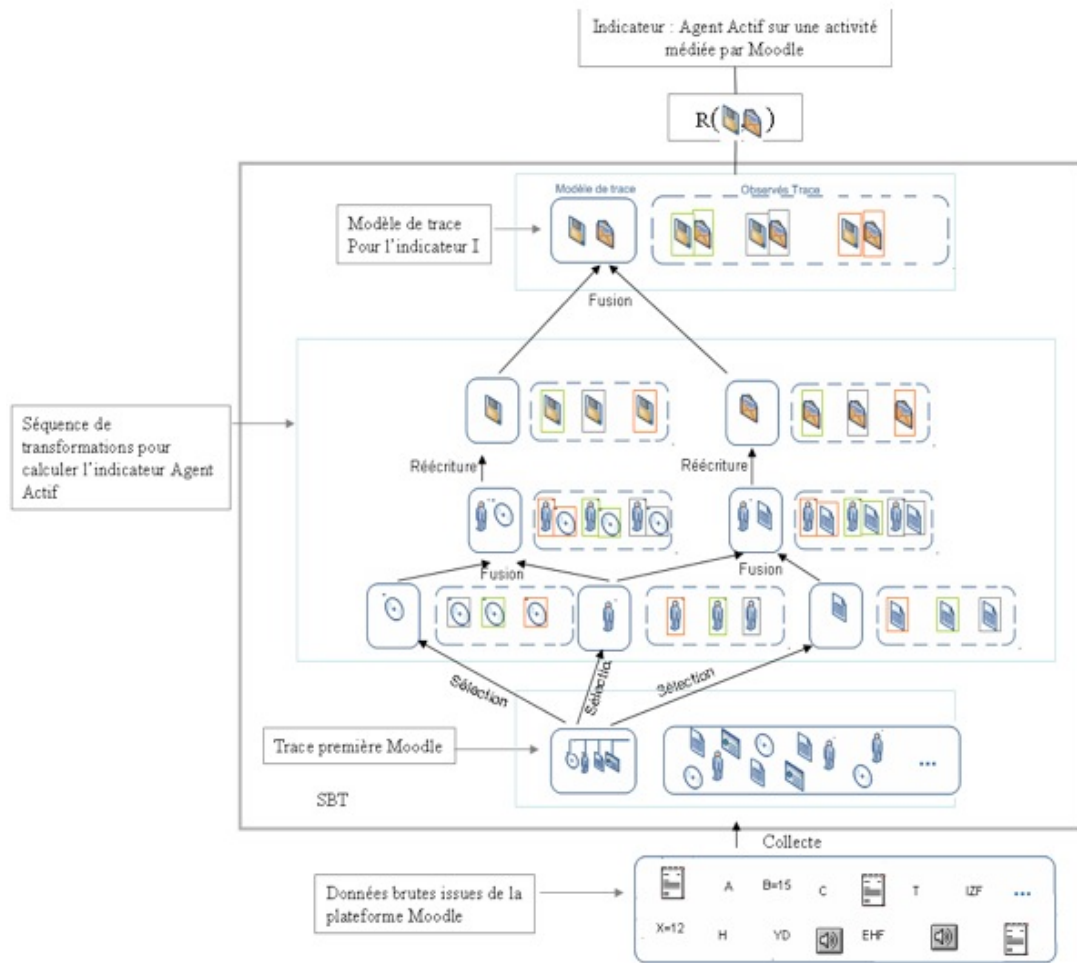


Figure 14 • Construction d’un nouvel indicateur dans un SBT, exemple : Agent actif pour une activité médiée par la plateforme Moodle.

2.3.2. Deuxième cas : calcul d’un nouvel indicateur à partir d’une séquence de transformations qui existe dans la bibliothèque de transformations pour la même plateforme visée.

Principe :

Il s’agit de calculer un indicateur supplémentaire sur le SBT, visant la même plateforme qu’un indicateur déjà défini dans le SBT. Cela signifie simplement que nous avons déjà construit des indicateurs dans le SBT et nous avons sauvegardé dans une bibliothèque les différentes transformations associées pour chaque modèle d’indicateur calculé. Le principe dans ce deuxième cas diffère du premier par la réutilisation d’une séquence de transformations définie pour un indicateur préexistant dans le SBT, et donc disponible dans la bibliothèque de transformations.

Exemple d’application : construire l’indicateur « Division du travail » sur Moodle sachant que nous disposons de l’indicateur « Agent Actif » sur Moodle.

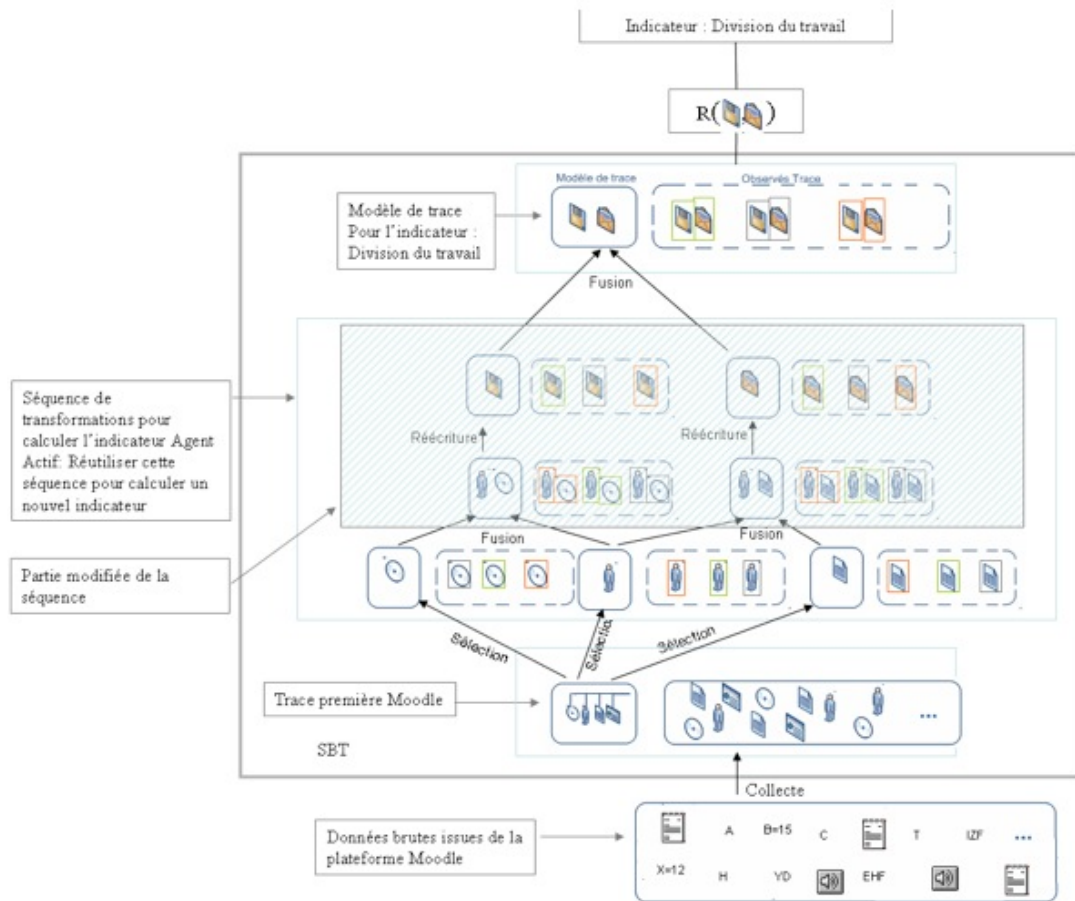


Figure 15 • Calcul d'un nouvel indicateur « division du travail » dans le SBT à partir d'une séquence qui existe dans la bibliothèque de modèles, issue de l'indicateur « Agent actif » par rapport à la trace première Moodle.

Etapes :

- Proposer un modèle de trace pour ce nouvel indicateur. Les instances de ce modèle de trace fournissent les données qui serviront d'entrée pour la règle de calcul définissant la valeur de l'indicateur.
- Chercher dans la bibliothèque des modèles d'indicateurs, un modèle de trace d'indicateur proche de ce nouveau modèle. Dans notre exemple, nous supposons que le modèle de l'indicateur « Agent actif », créé dans l'illustration du premier cas, est proche dans sa structure de celui de l'indicateur « Division du travail ». Les mesures de similarité entre les modèles sont un autre effort, que nous n'aborderons pas dans cet article.
- Modifier la séquence de transformations associée au modèle sélectionné. Cette modification permet de réutiliser la séquence de transformations associée à un modèle d'indicateur existant dans le SBT, pour déduire la transformation associée au nouvel indicateur visé. Le résultat de cette étape est une nouvelle séquence de transformations pour le nouvel indicateur, basée sur une séquence préalablement enregistrée dans le SBT.
- Exécuter la nouvelle séquence de transformations pour calculer les instances du nouveau modèle visé.

La figure 15 explique ces étapes. Nous pouvons remarquer que dans ce cas, nous ne modifions qu'une partie d'une ancienne séquence de transformations (la partie supérieure de la séquence, celle en zone hachurée dans la figure 15) pour déduire le nouveau modèle.

2.3.3. Troisième cas : Calcul d'un indicateur qui existe dans le SBT à partir d'une nouvelle plateforme d'apprentissage

Dans ce troisième cas, nous calculons un indicateur qui existe dans le SBT, mais visant une autre plateforme d'apprentissage. Comme le SBT est indépendant des plateformes d'apprentissage, nous

essayons cette fois de générer les valeurs d'un indicateur construit précédemment pour une autre plateforme d'apprentissage. Par exemple nous avons construit la séquence de transformations de l'indicateur « Agent Actif » sur la plateforme Moodle, et nous choisissons cette fois de reconstruire le même indicateur mais sur la plateforme WebCT. Dans ce cas, nous avons le modèle de l'indicateur que nous voulons calculer ainsi que sa séquence de transformations qui n'est pas réutilisable directement, et qui nécessite quelques modifications.

Etapes :

- Chercher le modèle de l'indicateur que l'on veut calculer dans la bibliothèque des modèles, par exemple le modèle de l'indicateur « Agent Actif ».
- Charger la séquence de transformations associée à cet indicateur.
- Comme la plateforme d'apprentissage change, nous savons que la trace première change, puisqu'elle est liée aux différentes activités offertes par la plateforme d'apprentissage. Ainsi, nous sommes devant l'obligation de modifier la partie inférieure de la séquence de transformations (du côté trace première) sans modifier toute la séquence. Comme le montre la figure 16, on ne modifie que la partie inférieure sélectionnée. Cette stratégie permet, grâce à un SBT, de s'adapter aux différentes plateformes d'apprentissage, ce qui donne la puissance de la méthode que nous proposons.

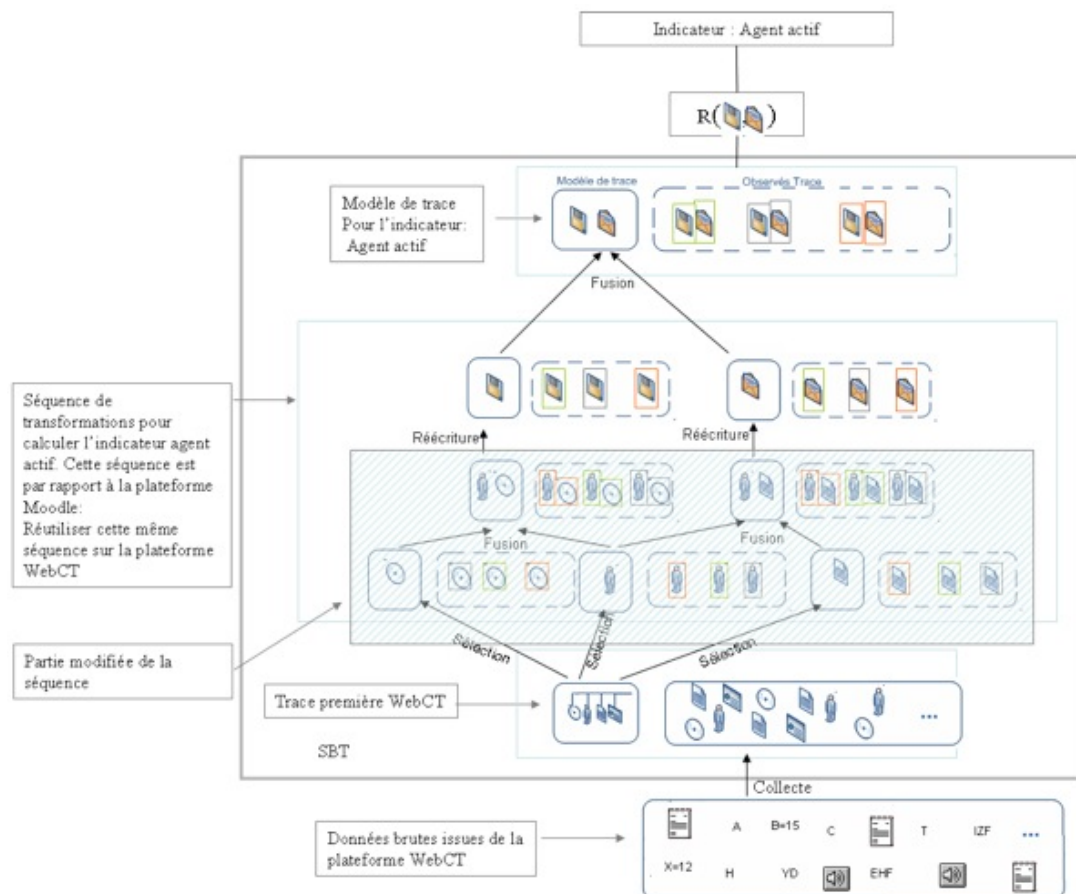


Figure 16 • Calcul d'un indicateur (« Agent actif » pour WebCT) à partir d'un indicateur existant (« Agent actif » pour Moodle) pour une nouvelle plateforme.

Nous avons présenté dans cette partie, une méthode pour calculer les indicateurs à partir des transformations des M-Traces basée sur l'ingénierie dirigée par les modèles de trace. Nous avons distingué trois possibilités : un nouvel indicateur dans une plateforme visée ; un nouvel indicateur (pour la même plateforme visée) que nous construisons à partir de la bibliothèque des modèles ; et un indicateur qui existe dans le SBT mais pour une autre plateforme d'apprentissage. Nous présentons dans la section suivante, une mise en œuvre de cette méthode sur une étude de cas bien précise sur la plateforme Moodle.

3. Mise en œuvre

Nous allons présenter dans cette troisième partie de l'article, l'implémentation des différentes étapes présentées dans les sections précédentes : le SBT, les opérateurs de transformations, et les séquences de transformations sur les modèles de trace. Le système à base de trace implémenté vise la plateforme Moodle (Moodle, 2008), mais son architecture est par nature ouverte aux autres plateformes d'apprentissage. Nous présentons dans les sections suivantes (1) la collecte des données en l'illustrant sur la plateforme Moodle pour générer une trace première, (2) les séquences de transformations de cette trace première avec des opérateurs génériques, et nous proposons des modèles pour quelques indicateurs. Nous expliquons enfin comment aboutir aux valeurs de ces indicateurs avec notre méthode.

3.1. La collecte pour les activités collaboratives

Nous proposons dans cette phase une collecte spécialisée visant les activités d'apprentissage collaboratif. Nous nous intéressons aux activités synchrones et asynchrones associant des acteurs de l'apprentissage partageant des ressources communes : Chat, Wikis, forums de discussion, tableaux blancs, éditeurs communs, et les échanges de type : messages privés.

3.1.1 Exemple sur Moodle

Dans un travail préliminaire (Djouad, 2008), nous avons défini un modèle pour une trace première Moodle. Nous avons amélioré et généralisé ce modèle pour les activités collaboratives utilisant les ressources que nous venons de citer. Il a été étendu pour prendre en compte les actions supplémentaires concernant la gestion des contacts représentée par les 4 dernières actions du modèle de la figure 17 pour étoffer les informations disponibles aux chercheurs qui souhaitent expliquer le comportement collaboratif des apprenants en s'appuyant sur les valeurs des indicateurs générés.

Nous proposons à l'utilisateur de notre outil la possibilité de choisir ce qu'il collecte selon ses besoins. Ce choix de la collecte dépend des modèles de trace que l'utilisateur souhaite construire pour calculer son indicateur. Par rapport à Moodle, l'outil se connecte à la base de données Moodle, importe les données nécessaires, et instancie le modèle de trace dans un format OWL qui respecte la syntaxe du parseur Jena (Jena, 2007).

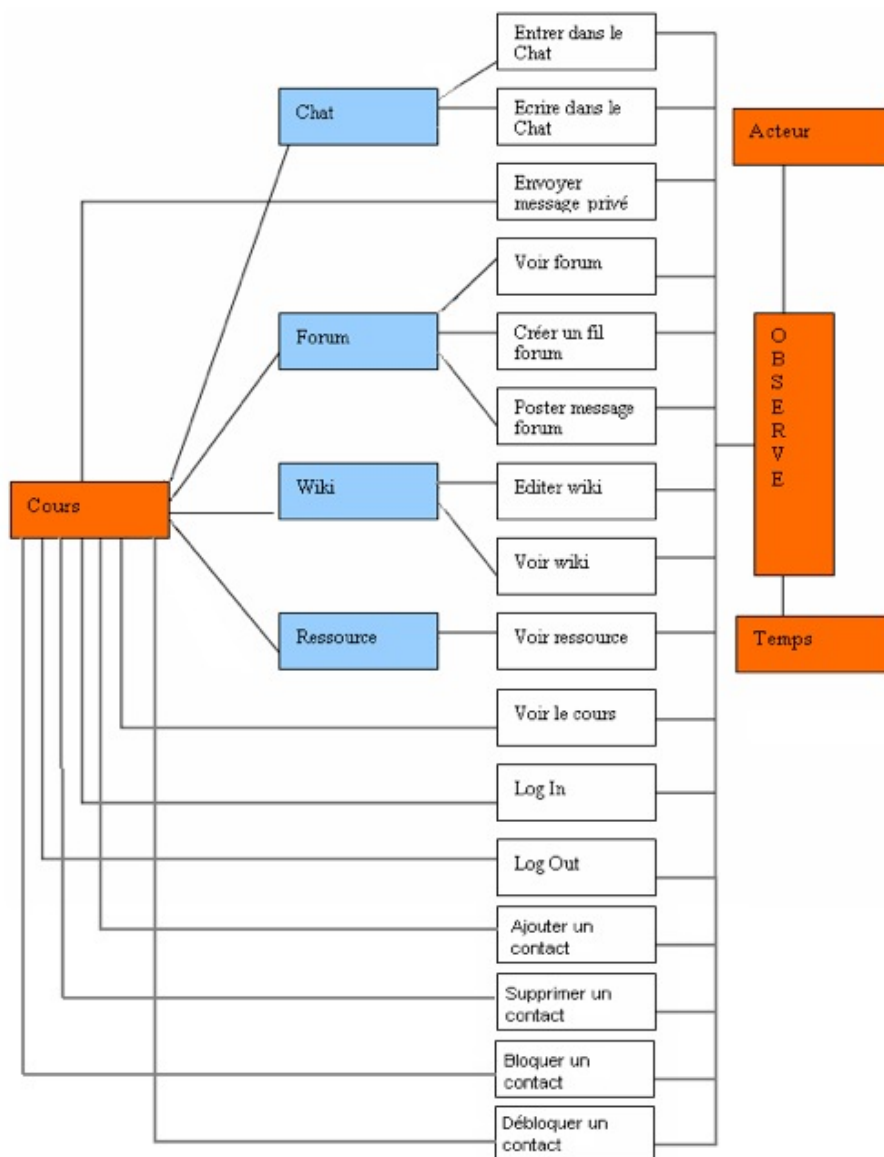


Figure 17 • Modèle de la trace première pour Moodle

Le choix de la collecte se fait avec une interface graphique (Fig. 18). Comme l'illustre la figure 18, la boîte de dialogue fournit la liste des actions à collecter. Ici, l'utilisateur choisit de ne collecter que 5 des 18 actions possibles. Les plugins de collecte récupèrent les données des sources de traçage etinstancient le modèle de la trace première. Le résultat de cette collecte est une trace OWL.

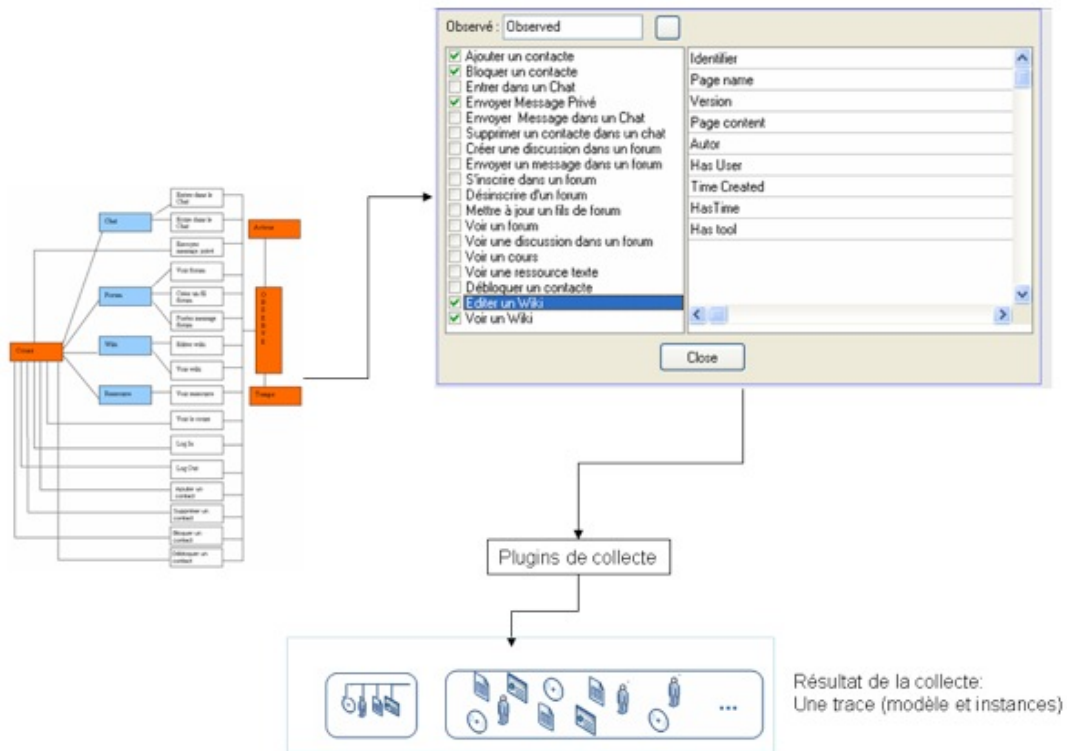


Figure 18 • Interface de choix des observables à collecter.

3.2. Les opérateurs de transformation du SBT implémenté

Nous proposons une collection d’opérateurs sur les traces permettant de faire des transformations génériques et indépendantes des plateformes de collaboration, puisque les opérateurs utilisent comme point de départ la trace première, qui est une abstraction des données de traçage issues des plateformes de collaboration. Nous classons ces opérateurs en trois classes présentées dans les sections 3.2.1, 3.2.2 et 3.2.3.

3.2.1. Des opérateurs qui ne modifient pas le modèle de trace

Ce sont les opérateurs qui ne modifient pas le modèle, mais seulement les instances de la trace. Cette classe regroupe les opérateurs suivants : le *matching*, la sélection et la fusion de 2 instances.

3.2.1.1. Le Matching

Cet opérateur identifie une séquence d’observés selon la structure d’un motif (une signature de tâche). La recherche de la signature est basée sur un algorithme proposé par (Mille et al., 1999). L’opérateur s’utilise de la manière suivante :

$TraceX := Matching(signatureA) [TraceY] ;$

Où *TraceX* est la trace résultante, *SignatureA* est la signature ou le motif utilisé pour définir le critère de notre recherche, tandis que *TraceY* est la trace source dans laquelle sont recherchés les épisodes correspondant à la signature. Par exemple, nous pouvons dire qu’une entrée *effective* dans un chat est une séquence d’actions de type : « entrer dans un chat : ChatEnter » suivie par « écriture dans cette ressource de chat : ChatWriteMessage ». La figure 19 illustre cet exemple de *matching* permettant d’identifier les entrées effectives à partir de la trace première à l’aide d’une signature de type (ChatEnter, ChatWriteMessage). Si par exemple A=ChatEnter et B= ChatWriteMessage, X et Y des observés de type quelconque, et si l’on considère la séquence d’observés $XYXYXYXYXYAYXYXAYXYXXBXYXYX$ en entrée avec un *matching* (A,B), la séquence résultat est alors AB.

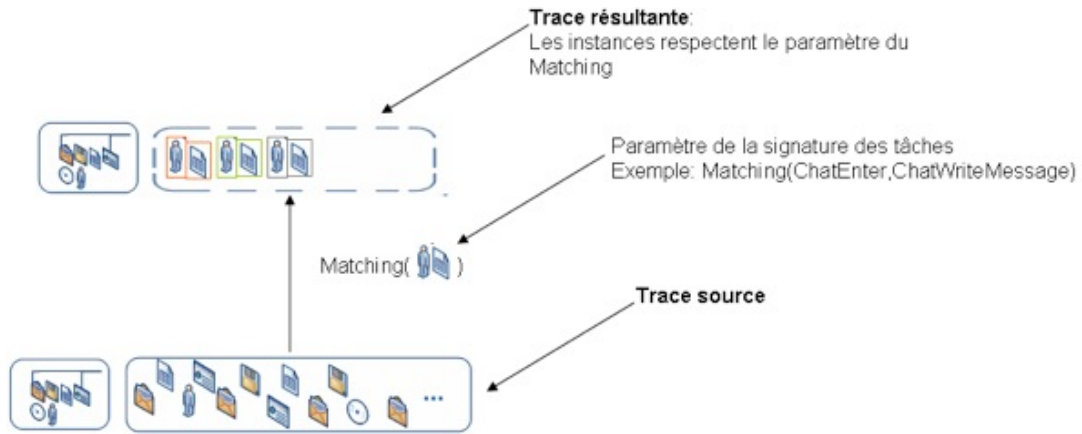


Figure 19 • Exemple d'un Matching

3.2.1.2. La sélection

Cet opérateur sélectionne une partie d'une instance. Les critères de la sélection sont : le temps, le type d'observé, l'outil associé à l'observé, ou l'acteur associé à l'observé. L'opérateur s'utilise de la manière suivante :

$TraceX := Sélection(Critère)[TraceY].$

Où *TraceX* est la trace résultante, *Critère* est une expression logique basée sur les attributs généraux d'une instance, tandis que *TraceY* est l'instance source dans laquelle sont sélectionnés les observés vérifiant le critère de sélection. La figure 20 illustre comment sélectionner tous les observés associés à un ActeurX à partir de la trace source :

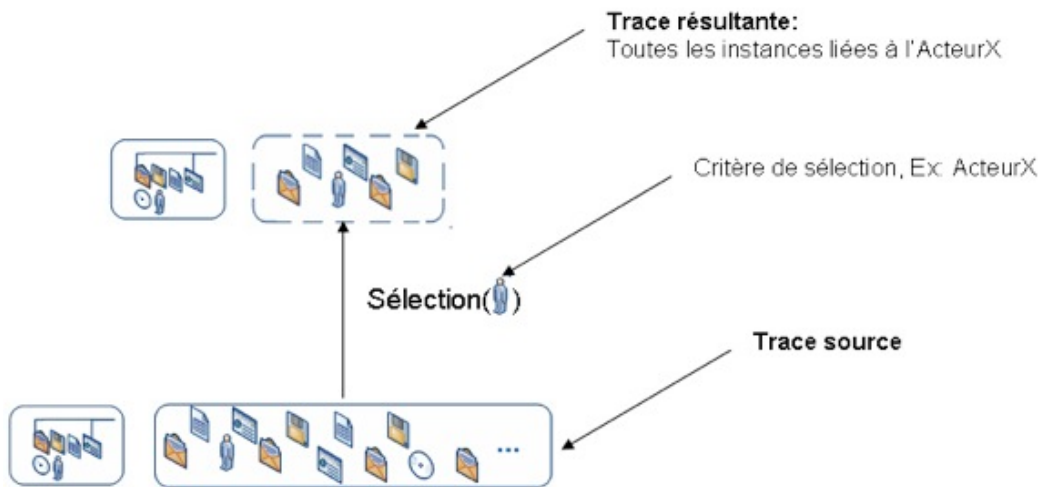


Figure 20 • Exemple d'une sélection

3.2.1.3. La fusion de deux instances de trace :

Cet opérateur concatène deux instances de traces sources (TraceA et TraceB) de même modèle dans une nouvelle instance (TraceC). La fusion est une union ensembliste des observés des deux traces. L'opérateur s'utilise de la manière suivante :

$TraceC := fusion(TraceA, TraceB)$

Pour illustrer l'utilisation de cet opérateur sur la figure 21, nous considérons d'une part les courriels envoyés et d'autre part les courriels enregistrés par un acteurX. Nous pouvons regrouper ces deux traces avec une fusion :

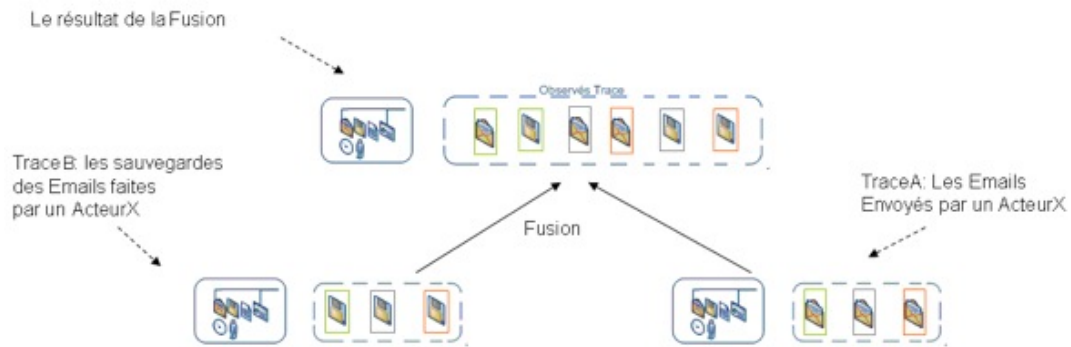


Figure 21 • Exemple d'une fusion de deux traces

3.2.2. Des opérateurs qui modifient un modèle de trace

Ces opérateurs modifient la structure du modèle. Le modèle de trace est structuré en plusieurs classes (Djouad, 2008). Nous proposons ici les deux opérateurs suivants : la *réécriture* du modèle et l'*élagage* du modèle.

3.2.2.1. La Réécriture d'une trace

Cet opérateur induit la création de nom de classe, ou d'un attribut qui appartient à une classe dans un modèle. La réécriture s'applique sur les instances en «produisant» une nouvelle trace reformulée conformément au modèle associé. L'opérateur s'utilise de la manière suivante :

Modele_X.NomClass1 :=Réécriture(ModeleX.NomClass2).

Cette opération est illustrée sur la figure 22 où l'on change le nom de la classe « Enregistrer » de l'ancien modèle d'une trace par le nom « Sauvegarder » :



Figure 22 • Exemple d'une réécriture

Cette *reformulation* est implantée pour l'instant de manière très simple, mais les travaux en cours sur les traces modélisées (Settouti et al., 2007) proposent un langage de requêtes permettant de réaliser des reformulations avec une richesse d'expression beaucoup plus grande. Ce langage plus riche sera implanté dans une prochaine version de l'opérateur de reformulation de notre SBT.

3.2.2.2. Élagage de modèle

Cet opérateur supprime des classes ou des attributs de classe dans un modèle et revient donc à reformuler la trace transformée dans une nouvelle trace réduite en expressivité (filtrage fin). On donne comme paramètres la liste des classes à conserver dans le modèle. L'opérateur s'utilise de la façon suivante :

TraceX :=Elagage(TraceY.(Liste des classes ou attributs conservés)).

La figure 23 donne un exemple d'élagage où seuls deux des attributs du modèle source sont conservés dans le modèle résultant.

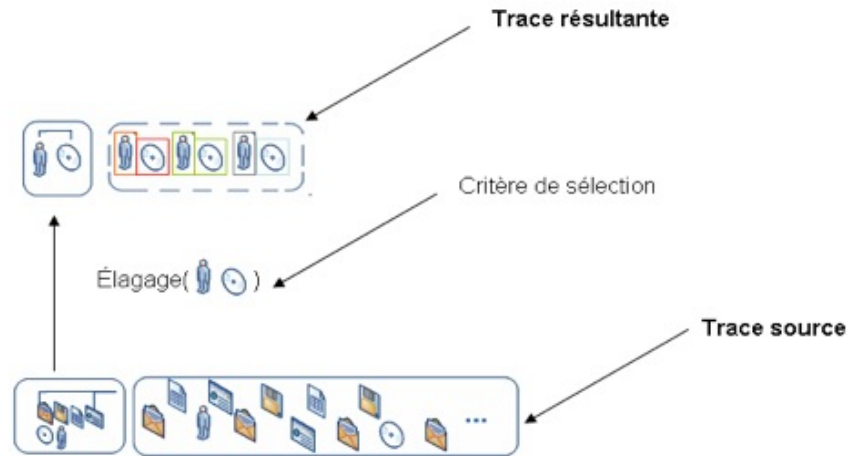


Figure 23 • Exemple d'un élagage

3.2.3. Les opérateurs orientés calcul d'indicateurs.

Nous proposons aussi deux opérateurs (Compter et Filtrer) utilisés pour calculer les indicateurs. Mais il ne s'agit plus de transformations car ces opérateurs ne sont pas dans le SBT.

Ils sont destinés à construire la fonction de calcul à partir de la trace transformée conforme au modèle associé à cet indicateur.

- **Compter** (Count) : compte le nombre des instances d'observés d'un certain type dans une trace. Le résultat est un nombre entier positif ou nul.

- **Filtrer** : permet de filtrer et trier un ensemble d'observés selon le temps, les types d'outils et/ou les types des observés.

Le lecteur intéressé par les détails de l'implémentation pourra suivre un exemple complet dans ([Djouad et al., 2008](#)).

4. Conclusions-Discussion

Nous proposons une méthode et des outils pour calculer les indicateurs d'activités collaboratives dans un EIAH disposant d'un système de gestion de bases de traces. La méthode que nous proposons est fondée sur l'ingénierie dirigée par les modèles (traces d'interactions modélisées), et s'appuie sur des séquences de transformations de modèles de trace et des instances associées permettant de collecter les observables nécessaires à un calcul explicite d'indicateurs d'activités collaboratives.

Au delà de la méthode, nous proposons une architecture et un outil permettant de construire et gérer les traces d'interaction nécessaires au calcul d'indicateurs et, c'est déterminant pour l'ingénierie proposée, d'accompagner le processus de modélisation en facilitant la réutilisation des modèles aussi bien pour la collecte des observables que pour le calcul des indicateurs eux-mêmes. Nous montrons l'intérêt de cette méthode et de ce qu'elle apporte par rapport au calcul ad-hoc des indicateurs, par le caractère déclaratif du calcul d'indicateur à un niveau abstrait tout en effectuant les opérations conformes à ces déclarations en situation d'apprentissage.

À l'image de ce qui est proposé par ([Dimitracopoulou, 2008](#)), notre méthode est largement indépendante de la plateforme cible. En effet, elle *décrit et opérationnalise* le calcul d'un nouvel indicateur à partir des séquences de transformations de la M-Trace première ignorant la plateforme cible. Seule la collecte initiale des événements est réalisée spécifiquement pour la plateforme cible, fournissant la trace modélisée dite *première*, racine de l'arbre des transformations débouchant sur les *feuilles* de calcul d'indicateurs d'activités collaboratives. L'utilisateur peut ainsi réutiliser des séquences de transformations pour calculer les mêmes indicateurs pour d'autres plateformes d'apprentissage assurant l'indépendance entre la plateforme d'apprentissage et la manière de calculer l'indicateur d'activité collaborative. L'utilisation et la réutilisation des séquences de transformations appliquées sur les M-Traces est le point fort et original de la méthode que nous proposons.

Cette méthode s'applique aussi bien pour les systèmes sociaux qui impliquent un grand nombre de participants que pour des activités de collaboration en petits groupes (de 2-3 personnes) ; la notion de trace modélisée première est générique et ne limite pas le nombre d'apprenants pris en compte (il faut simplement que les apprenants disposent d'un identificateur puisque nous partons d'une M-Trace première collectant les événements issus d'un nombre d'apprenants quelconque).

Elle peut naturellement s'appliquer pour calculer les indicateurs d'activités individuelles, mais un volet important de notre travail consiste à étudier les indicateurs d'activités collaboratives proposés dans la littérature et à définir, pour chacun d'eux, les modèles abstraits des traces à collecter pour en permettre le calcul. Une bibliothèque d'indicateurs avec leurs modèles de trace associés est donc en cours de constitution, spécifiquement pour les indicateurs d'activités collaboratives. Publier une telle bibliothèque intéresse tous les concepteurs d'EIAH souhaitant les intégrer, et de plus si, comme nous le proposons pour Moodle, il existe un module de collecte associé à la plateforme cible, l'implémentation sera rapide et pourra bénéficier de l'opérationnalisation automatique des transformations de M-Traces et du calcul final de l'indicateur. Ainsi, notre approche peut en extension générer *les données nécessaires* pour produire des indicateurs qui analysent le contenu d'interactions *génériques*.

Par exemple, nous pouvons générer la M-Trace de type « Chat » d'un apprenant, et utiliser un opérateur de type *Find(event, {list of properties})* qui compte combien de fois le mot "nous" a été écrit dans l'instance de cette M-Trace. De tels opérateurs peuvent être ajoutés facilement dans la classe des opérateurs orientés calcul des indicateurs proposés dans la section 3.2.3.

Nous planifions de tester notre SBT implémenté ainsi que la bibliothèque de transformations de modèles sur d'autres plateformes d'apprentissage que Moodle. Nous prévoyons aussi d'ajouter d'autres opérateurs de transformation pour donner plus de richesse au mécanisme de transformation proposé par notre méthode, en particulier l'opérateur de réécriture, puissant outil d'abstraction de traces d'interaction.

¹ Les traces transformées peuvent être reconstruites dynamiquement en appliquant les règles de transformation. Il n'est donc pas forcément nécessaire de répliquer les observés qui ne font pas l'objet d'une transformation.

BIBLIOGRAPHIE

AVOURIS, N., MARGARITIS, M., KOMIS, V., MELENDEZ, R., SAEZ, A. (2003). ModellingSpace: Interaction Design and Architecture of a collaborative modelling environment. *In proceedings of the 6th conference of Computer Based Learning in Science (CBLIS)*, Nicosia, Cyprus, p. 993-1004. Disponible sur Internet : http://cblis.utc.sk/cblis-cd-old/2003/4.PartC/Papers/Learning_Env_Des_Modelling_Activities/Avouris.pdf

BÉZIVIN, J., GERBÉ, O. (2001). Towards a precise Definition of the OMG/MDA framework. *In Proceedings of the 16th Conference on Automated Software Engineering IEEE (ASE'2001)*, San Diego (USA), p 273-280. Disponible sur Internet : <http://www.sciences.univ-nantes.fr/lina/atl/www/papers/ASE01.OG.JB.pdf>

DIAGNE, F. (2008). Em-AGIIR : un Environnement Multi-AGent ouvert pour la supervision à partir d'Indicateurs Réutilisés. *2ième rencontre des jeunes chercheurs RJC-EIAH08*, Lille, France, p. 65-70.

CHOQUET, C., IKSAL, S. (2007). Modélisation et construction de traces d'utilisation d'une activité d'apprentissage : une approche langage pour la réingénierie d'un EIAH. *Revue STICEF numéro spécial*, Vol. 14. Disponible sur Internet : http://sticef.univ-lemans.fr/num/vol2007/14-choquet/sticef_2007_choquet_14.htm

DIMITRACOPOULOU, A., BRUILLARD, E. (2006). Enrichir les interfaces de forums par la visualisation d'analyses automatiques des interactions et du contenu. *Revue STICEF*, Vol. 13. Disponible sur Internet : http://sticef.univ-lemans.fr/num/vol2006/dimitracopoulou-10/sticef_2006_dimitracopoulou_10.htm

DIMITRACOPOULOU, A. (2004). State of the art of interaction Analysis: Interaction Analysis Indicators. Interaction and Collaboration Analysis supporting Teachers' and Students' Selfregulation (ICALTS) JEIRP Deliverable D.26.1.1. Kaleidoscope NoE, December 2004. 153 p. Disponible sur Internet : <http://telearn.no-kaleidoscope.org/open-archive/file?Dimitrakopoulou-Kaleidoscope-2004.pdf>

- DIMITRACOPOULOU, A. (2008). Computer based Interaction Analysis Supporting Self-regulation: Achievements and Prospects of an Emerging Research Direction. In Kinshuk, M.Spector, D.Sampson, P. Isaías (Guest editors). Technology, Instruction, Cognition and Learning (TICL). Disponible sur Internet : http://ltee.org/adimitr/wp-content/uploads/2008/10/dimitracopoulou_ticl_special_issue_final_draft.pdf
- DJOUAD, T. (2008). Analyser l'activité d'apprentissage collaboratif : Une approche par transformations spécialisées de traces d'interactions. *2ième rencontre des jeunes chercheurs RJC-EIAH08*, Lille, France, p. 93-98. Disponible sur Internet : <http://liris.cnrs.fr/Documents/Liris-3604.pdf>
- DJOUAD, T., MILLE, A., REFFAY, C., BENMOHAMED, M. (2008). Calcul des indicateurs collaboratifs à partir des transformations spécialisées dans un SBT. Rapport de recherche 2008. Disponible sur Internet : <http://liris.cnrs.fr/~tdjouad/Website/rapport.pdf>
- Distributed Management Task Force (1999). Common Information Model (CIM) Specification. Rapport technique, DSP0004. Disponible sur Internet : <http://www.dmtf.org/standards/documents/CIM/DSP0004.pdf>
- INES (2008). Guide d'utilisation de la plateforme INES. Disponible sur internet : <http://pf-ma.uvt.rnu.tn/foad/guide/index.htm>
- JAILLET, A. (2005). Peut-on repérer les effets de l'apprentissage collaboratif à distance. *Distances et savoirs*, Vol. 3 n°1, p. 49-66.
- JERMANN, P.-R. (2004). Computer Support for Interaction Regulation in Collaborative Problem-Solving. *Thèse de doctorat*, Genève. Disponible sur Internet : <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.122.2069&rep=rep1&type=pdf>
- JENA (2007). *A Semantic Web Framework for Java*, <http://jena.sourceforge.net/>
- LAVALLARD, A. (2008). Exploration interactive d'archives de forums : Le cas des jeux de rôle en ligne. *Thèse de doctorat*, Université de Caen, soutenue le 2 juillet 2008. Disponible sur Internet : <http://tel.archives-ouvertes.fr/tel-00292617/en/>
- MARTÍNEZ, A., DIMITRIADIS, Y., GÓMEZ, E., RUBIA, B., DE LA FUENTE, P. (2003). Combining qualitative and social network analysis for the study of classroom social interactions. *Computers and Education*, Vol. 41 n°4, p.353-368. Disponible sur Internet : http://www.infor.uva.es/~amartine/research/pubs-2003/CompEduc_martinez03.pdf
- MAY, M. (2008). Une approche pour tracer finement les communications médiatisées en situation d'apprentissage. *2ième rencontre des jeunes chercheurs RJC-EIAH08*, Lille, France, p. 71-76. Disponible sur Internet : http://noce.univ-lille1.fr/rjc_eiah2008/actes.pdf
- MERCERON, A., YACEF, K. (2004). Mining Learner Data Captured from a Web-based Tutoring Tool: Initial Exploration and Results. *Journal of Interactive Learning Research, Special Issue on Computational Intelligence in Web-Based Education*, Vol 15 n° 4, p.319-346.
- MILLE, A., PRIÉ, Y. (2006). Une théorie de la trace informatique pour faciliter l'adaptation dans la confrontation logique d'utilisation/logique de conception. In *the Proceeding of : la 13èmes journées de Rochebrune - Traces, Enigmes, Problèmes*, Rochebrune, France. Disponible sur Internet : <http://liris.cnrs.fr/Documents/Liris-3002.pdf>
- MILLE, A., FUCHS, B., CHIRON, B. (1999). Raisonnement fondé sur l'expérience : un nouveau paradigme en supervision industrielle. *Revue d'intelligence artificielle*, Vol. 13, p.97-128.
- MOSTOW, J., BECK, J., CUNEO, A., GOUVEA, E., HEINER, C. (2005). A Generic Tool to Browse Tutor-Student Interactions: Time Will Tell! In *Proceedings of the 12th International Conference on Artificial Intelligence in Education (AIED 2005)*, Amsterdam, p.29-32.
- Moodle (2008). Documentation de la plateforme MOODLE : *Modular Object-Oriented Dynamic Learning Environment*. <http://docs.moodle.org/fr/Accueil>
- NODENOT, T. (2005). Contribution à l'ingénierie dirigée par les modèles en EIAH : le cas des situations-problèmes coopératives. *Habilitation à Diriger des Recherches en Informatique*, Laboratoire d'Informatique de l'Université de Pau et des Pays de l'Adour. Disponible sur Internet : <http://idee.iutbayonne.univ-pau.fr/~nodenot/HDR/Habilitation-t-nodenot.pdf>
- OMG (2002). *Meta-Object Facility (MOF) Specification Version 1.4*. Object Management Group. <http://www.omg.org/technology/documents/formal/mof.htm>
- PHAM THI NGOC, D. (2008). Réingénierie des EIAH : automatiser et réutiliser le savoir-faire en analyse d'usage. *2ième rencontre des jeunes chercheurs RJC-EIAH08*, Lille, France, p. 99-104.
- REFFAY, C., LANCIERI, L. (2006). Quand l'analyse quantitative fait parler les forums de discussion, *Revue STICEF, recueil 2006, numéro spécial forum de discussion en éducation*, p.255-288. Disponible sur internet : http://sticef.univ-lemans.fr/num/vol2006/reffay-13/sticef_2006_reffay_13.htm
- SANTOS, O.-C., RODRÍGUEZ, A., GAUDIOSO, E., BOTICARIO, J.-G. (2003). Helping the tutor to manage a collaborative task in a web-based learning environment. *Communication in the Workshop: Towards Intelligent Learning Management Systems, AIED (Artificial Intelligence in Education) international conference*, Sydney, Australia. p.72-81.

SEIDWITZ, E. (2003). What models Mean. *IEEE Software*, Vol.20 n° 5, p.26-32.

SETTOUTI, L.-S., PRIÉ, Y., MILLE, A., MARTU, J.-C. (2006). Systèmes à base de traces pour l'apprentissage humain. *Communication in the international TICE, Technologies de l'Information et de la Communication dans l'Enseignement Supérieur et l'Entreprise*, Toulouse, France.

SETTOUTI, L.-S., PRIÉ, Y., MARTY, J.-C., MILLE, A. (2007). Vers des Systèmes à Base de Traces modélisées pour les EIAH. *Rapport de recherche RR-LIRIS-2007-016*. Disponible sur internet : <http://liris.cnrs.fr/Documents/Liris-2882.pdf>

SOLLER, A., MARTINEZ, A., JERMANN, P., MUEHLENBROCK, M. (2005). From Mirroring to Guiding: A Review of State of the Art Technology for Supporting Collaborative Learning. *IJAIED: International Journal of Artificial Intelligence in Education* vol. 15, p. 261-290. Disponible sur Internet : http://aied.inf.ed.ac.uk/members05/archive/Vol_15/Soller/Soller05.html

TEDESCO, P.-A. (2003). MARCo: Building an Artificial Conflict mediator to Support Group Planning Interactions. *International Journal of Artificial Intelligence in Education* Vol.13, p.117-155.

VOISIN, J., VIDAL, P. (2007). Une approche conduite par les modèles pour le traçage des activités. *Revue STICEF numéro spécial : Analyses des traces d'utilisation dans les EIAH.*, Vol 14. Disponible sur Internet : http://sticef.univ-lemans.fr/num/vol2007/17-broisin/sticef_2007_broisin_17.htm

WBEM (1999). Web Based Enterprise Management. Distributed Management Task Force, Inc. Disponible sur Internet : <http://www.dmtf.org/standards/wbem/>

■ A propos des auteurs

Tarek DJOUAD est doctorant en informatique depuis novembre 2006. Il prépare une thèse en cotutelle entre l'université Claude Bernard Lyon1, laboratoire LIRIS (Laboratoire d'InfoRmatique en Image et Systèmes d'information) et l'université MentouriConstantine, Algérie, Laboratoire LIRE(Laboratoire en Informatique REpartie). Il s'intéresse au calcul des indicateurs d'activités collaboratives par transformation de modèles de trace d'interaction.

Adresse : Equipe SILEX (Supporting Interactions, Learning by EXperience), Bâtiment Nautibus (710), 43, bd du 11 novembre 1918 , 69 622 Villeurbanne Cedex

Courriel : tarek.djouad@liris.cnrs.fr

Toile : <http://liris.cnrs.fr/membres?idn=tdjouad>

Alain MILLE est professeur à l'Université Claude Bernard Lyon1. Responsable de l'équipe SILEX (Supporting Interactions and Learning from EXperience), il est un spécialiste du raisonnement à partir de l'expérience tracée. Cette théorie est appliquée avec succès à l'ingénierie des connaissances, à l'assistance aux activités médiées par un environnement informatique, en particulier les activités d'apprentissage médiées par un environnement informatique pour l'apprentissage humain.

Adresse : Equipe SILEX (Supporting Interactions, Learning by EXperience), Bâtiment Nautibus (710), 43, bd du 11 novembre 1918 , 69 622 Villeurbanne Cedex

Courriel : alain.mille@liris.cnrs.fr

Toile : <http://liris.cnrs.fr/alain.mille/>

Christophe REFFAY est maître de conférence en informatique et chercheur en détachement à l'Institut National de la Recherche Pédagogique , au laboratoire "Sciences Techniques Éducation Formation" de l'ENS de Cachan. Il s'intéresse à l'analyse des interactions dans les situations collaboratives de formations/apprentissage à distance et en ligne et en particulier aux modèles issus des réseaux sociaux. Depuis 2006, il s'intéresse aux problèmes du recueil, de l'indexation et de la structuration et de l'anonymisation des données issues de telles expérimentations, en vue d'en constituer des corpus échangeables avec des chercheurs extérieurs à l'expérimentation.

Adresse : UMR STEF Bâtiment Cournot, ENS de Cachan, 61 avenue du Président Wilson, F-94235 Cachan Cedex

Courriel :

Toile : Christophe.Reffay@inrp.fr <http://lifc.univ-fcomte.fr/~reffay/>

Mohamed BENMOHAMMED est professeur en informatique à l'université Mentouri Constantine Algérie, au laboratoire LIRE (Laboratoire en Informatique REpartie) depuis 1998. Il s'intéresse aux architectures parallèles des systèmes informatiques, et aux systèmes distribués.

Adresse : Université Mentouri Constantine, B.P. 325 Route Ain El Bey, Constantine, 25017, Algérie

Courriel : ben_moh123@yahoo.com

Référence de l'article :

Tarek DJOUAD, Alain MILLE (LIRIS, Lyon), Christophe REFFAY (LIFC, Besançon), Mohamed BENMOHAMED (LIRE, Algérie), Ingénierie des indicateurs d'activités à partir de traces modélisées pour un Environnement Informatique d'Apprentissage Humain, *Revue STICEF*, Volume 16, 2009, ISSN : 1764-7223, mis en ligne le 23/11/2009, <http://sticef.org>

© Revue Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation, 2009