



Modélisation et utilisation de traces d'expérience d'utilisation de systèmes informatiques

Yannick Prié
Equipe SILEX
LIRIS – Université Claude Bernard Lyon 1

M2R Informatique
2006-2007



Plan

- Introduction : pourquoi les traces ?
- Approche Musette
- Systèmes à base de traces
- Exemples de travaux
- Conclusion



Plan

- **Introduction : pourquoi les traces ?**
- Approche Musette
- Systèmes à base de traces
- Exemples de travaux
- Conclusion



Constat

- Pratiques et usages des outils informatiques en mutation
 - massification des usages
 - tâches de plus en plus fondées sur des outils informatiques
 - outils de plus en plus complexes et interconnectés
- Tâches et utilisation de ressources
 - accès croissant aux « ressources » informatiques via des « portails » plutôt que par des applications spécialisées
 - intégration et mobilisation de ressources variées pour des tâches faiblement spécifiées et difficiles à assister
 - « évidence » de l'effet de contexte dans les usages
- Liens conception / usages
 - conception des outils (fatalement) en décalage avec l'utilisation
 - tension entre simplicité d'utilisation et adaptabilité aux besoins dans les pratiques



Volonté

- Objectif général
 - construire des systèmes qui fonctionnent en intelligence avec leurs utilisateurs.
 - interfaces graphiques / assistants
 - comprendre et analyser les usages
 - outils conceptuels et informatiques pour l'analyse
- Approche générale
 - utiliser l'expérience d'utilisation des systèmes
 - pour améliorer leur fonctionnement, faciliter les tâches des utilisateurs qui les utilisent
 - pour analyser celle-ci *a posteriori* en tant que trace de l'usage
 - expérience
 - traces concrètes d'interaction entre système et utilisateur



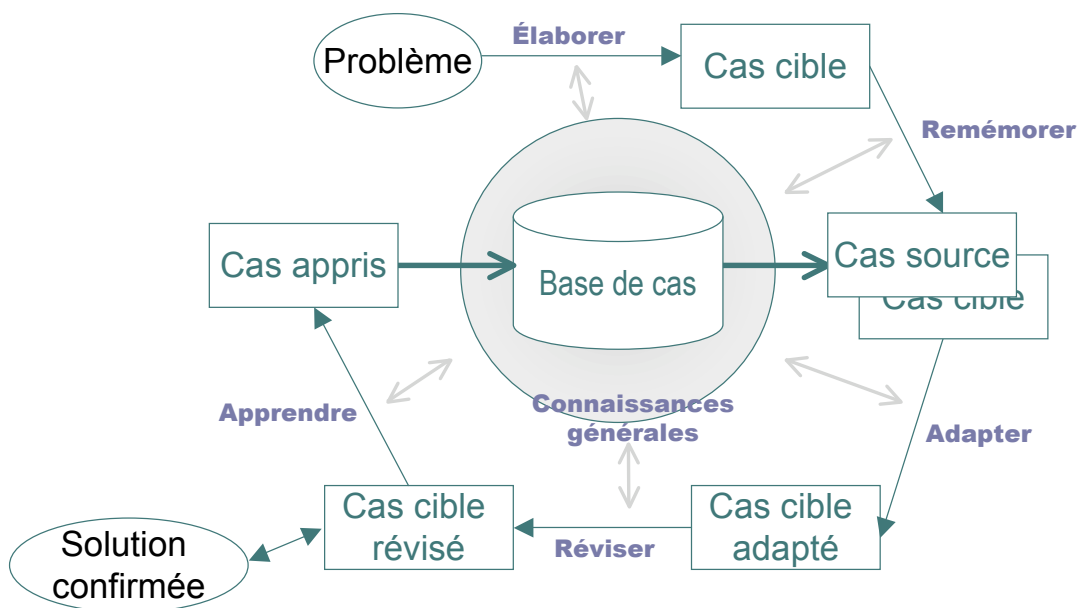
Historique / CEXAS

- Mise en place de l'approche MUNETTE
 - Modéliser les Usages et les Tâches pour Tracer l'Expérience
 - Passage du RàPC au RàPE
 - Attention à la trace en tant que telle
- Plus récemment : SBT
 - Systèmes à Base de Traces
 - Généralisation
 - Volonté de mettre en place une « théorie de la trace informatique »
- Projets et applications variés

● ● ● | Du RàPC *au RàPE*

- Origine équipe
 - construction de différents systèmes fondés sur le RàPC
- Raisonement à partir de cas
 - un cas représente une instance concrète d'un problème et de sa solution
 - idée
 - stocker des cas sous la forme pb/solution
 - réutiliser ces cas comme solutions partielles pour de nouveaux problèmes
- Cycle du raisonnement à partir de cas
- Variantes et améliorations diverses

● ● ● | Le cycle du RàPC





Du RàPC au RàPE

- Systèmes RàPC « canoniques »
 - tâches bien fixées
 - trouver conditions tel que résultat_souhaité
 - base de cas = { [conditions → résultat] }
- Systèmes RàPC « non canoniques »
 - tâches plus génériques
 - supervision, recherche d'information
 - base de cas = ensemble de tableaux de bord utiles, parcours sur le web
 - utilisateur au centre du système d'assistance
 - modèles de connaissances évolutifs, cas non complètement spécifiés

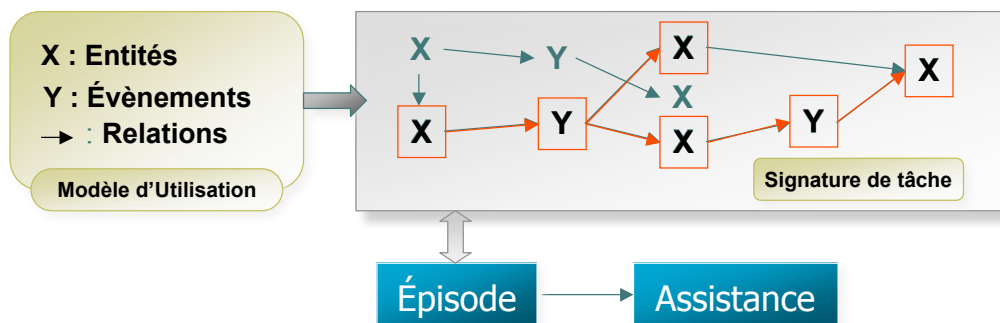
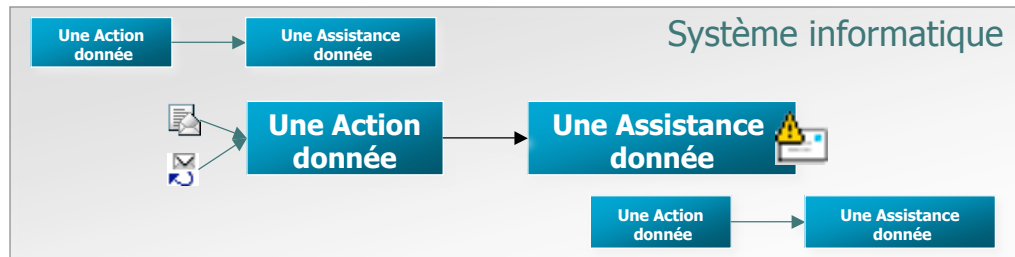


Du RàPC au RàPE

- Limitations du RàPC
 - structure des cas définie a priori, figée
 - connaissances d'explication acquises à l'avance
 - finalement
 - faibles possibilités d'adaptation au contexte, à l'utilisateur
- Idée
 - se donner la possibilité de définir des cas en fonction des besoins et des contextes d'utilisation, dont la liste n'est jamais figée
- Pour cela
 - enregistrer une trace correspondant à une tâche générique d'utilisation du système
 - pouvoir y considérer à volonté des épisodes d'utilisation



Du RàPC au RàPE



Du RàPC au RàPE

- RàPE
 - un cas est un épisode découpé dans la trace
 - un problème peut s'exprimer en fonction de ce qu'a déjà fait l'utilisateur
 - assistance : appel à l'aide, assistance fondée sur l'expérience des épisodes précédents
- Concepts
 - modèle d'utilisation
 - ensemble des descripteurs d'un système informatique
 - trace d'utilisation
 - instances des descripteurs issus de l'utilisation du système
 - épisode
 - sous-partie de la trace correspondant à la signature d'une tâche particulière



Attention à la trace en tant que telle

- Emergence de l'observateur
 - le modèle d'utilisation n'est pas absolu, il dépend de ce que choisit l'observateur
 - une application peut avoir plusieurs modèles d'utilisation (qui diffèrent d'un éventuel modèle de conception)
 - tout dépend de ce que veut faire le modélisateur de l'expérience dans son système
- Trace primitive
 - niveau fondamental à la charnière entre le concepteur du système fondé sur l'expérience et l'utilisateur
 - attention particulière à lui porter
- La trace pour l'utilisateur
 - l'utilisateur a une pratique courante de la trace, en tant qu'inscription
 - pas forcément besoin de cas



Plan

- Introduction : pourquoi les traces ?
- **Approche Musette**
- Systèmes à base de traces
- Exemples de travaux
- Conclusion



Exemple : Human-Links

- Outil pour la veille technologique
- Gestion complexe d'un espace documentaire
 - Aide à la collecte, au traitement, et à la diffusion d'informations glanées
 - Documents
 - Représentés par un profil (ensemble de mots-clés)
 - Regroupés en catégories (avec profils)
 - Requêtes (distribuées ou non)
 - Contact
- Visualisation cartographique des documents et des catégories
 - Répartition en fonction de l'attraction d'un document vers une catégorie



Logiciel Human-Links

Hiérarchie de catégories

Catégorie courante

Profil en mots-clés de la catégorie courante

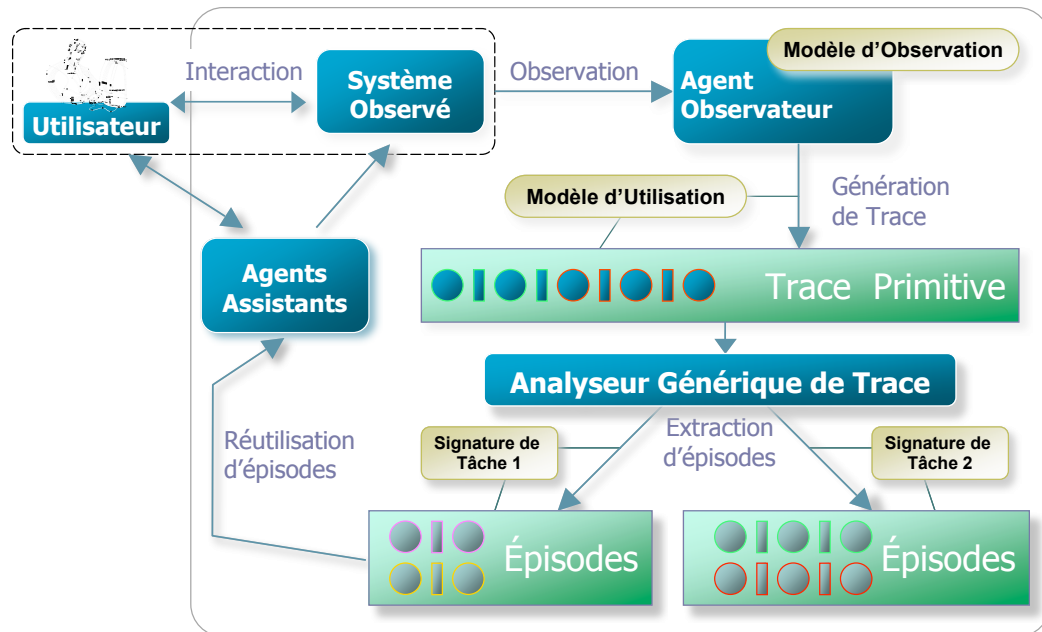
Carte thématique

Item document

Liste détaillée d'items

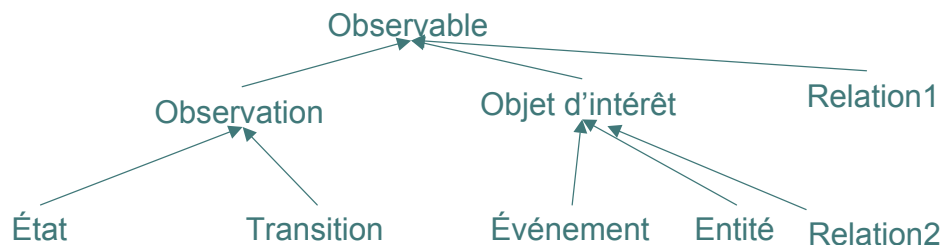


MUSETTE : les grandes étapes



MUSETTE-Base

- « top level ontology » = ensemble de classes à spécialiser en un modèle d'utilisation



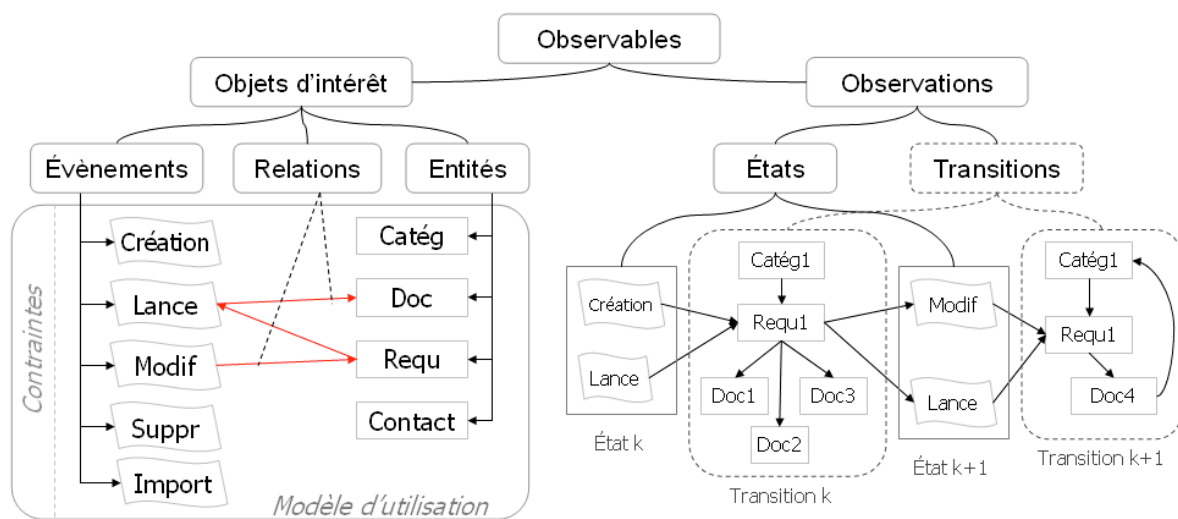
○ Contraintes

- Ordre séquence état/transition
- Etat contient entités
- Transition contient Evénements
- Relations entre objets d'intérêt

● ● ● | Modèle d'utilisation

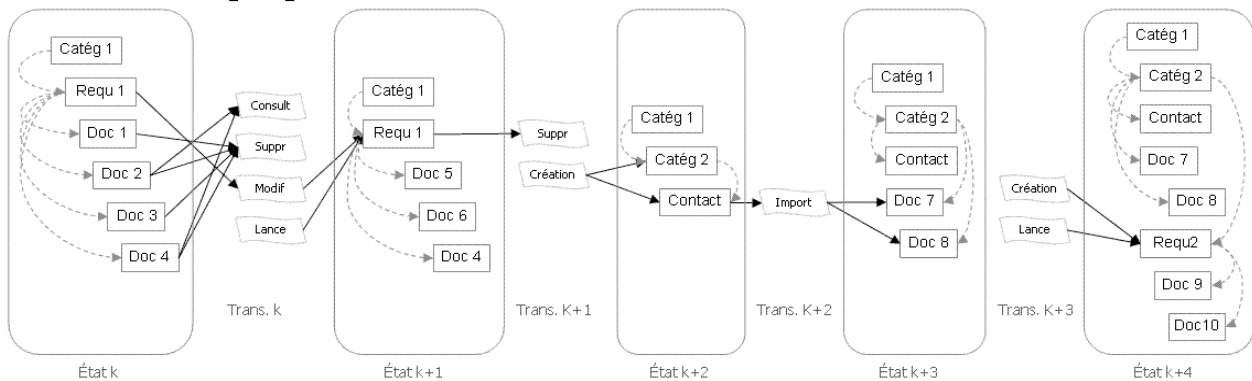
- Ensemble de types d'entités, de types d'événements et de types de relations
- Dans la mesure où le langage le permet
 - contraintes sur les types (spécialisation, exclusion mutuelle...)
 - contraintes sur les relations (domaine et co-domaine, transitivité, relations inverses, ...)
 - contraintes sur la disposition des objets d'intérêt dans les observations

● ● ● | MU simplifié et exemple de trace





Trace plus longue



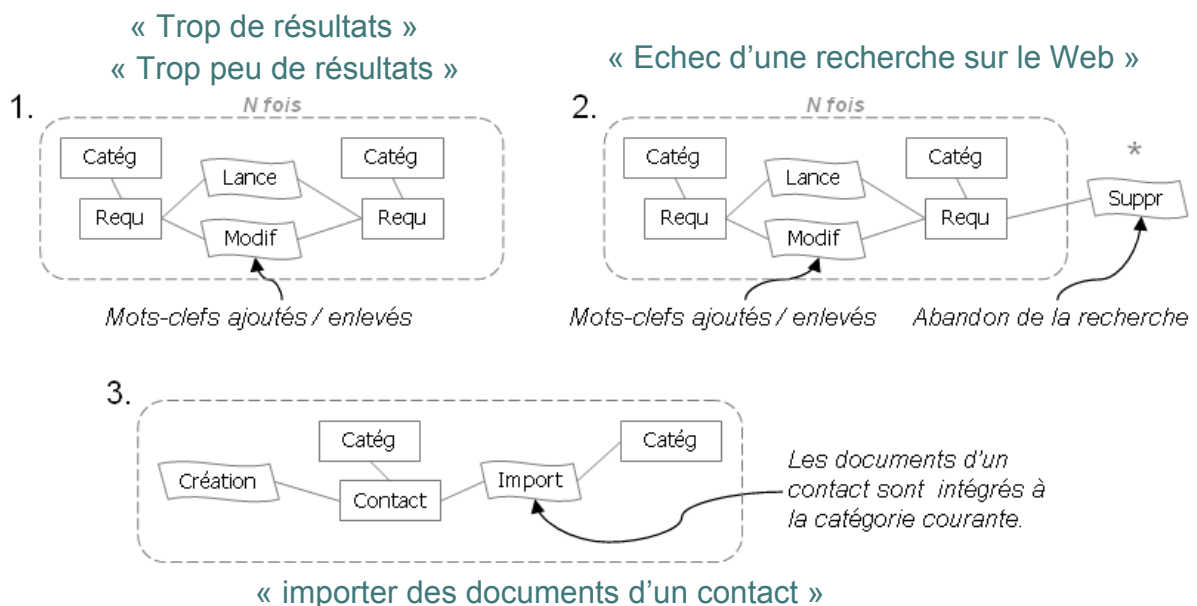
Signatures de tâches expliquées et épisodes

- Le modèle d'utilisation permet d'inscrire l'utilisation dans une trace primitive
- La trace contient potentiellement des épisodes d'utilisation re-traçant une expérience utilisable pour l'assistance en contexte
- Les épisodes sont repérés dans la trace grâce à des signatures de tâches « expliquées »
- Une signature s'instancie dans la trace et définit un épisode

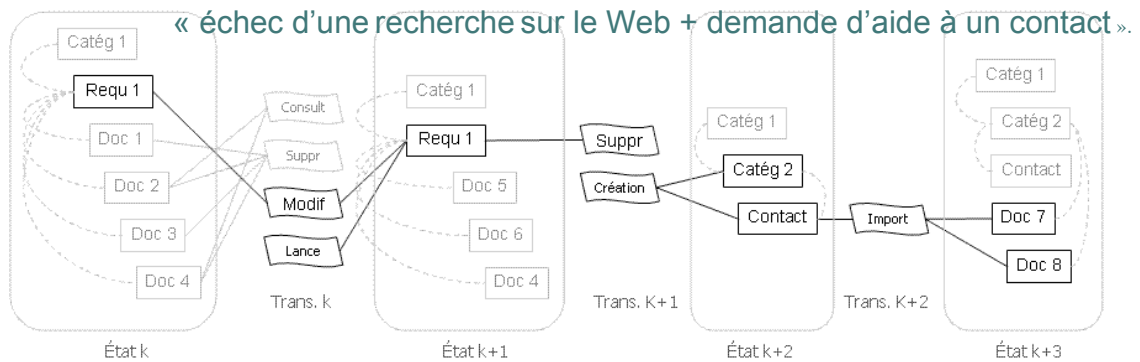
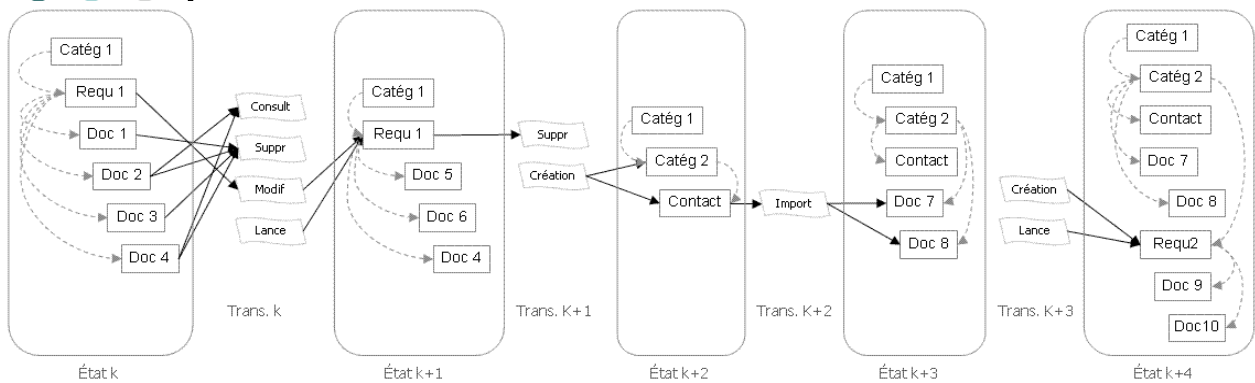
Signature de tâches expliquées

- Composition
 - motif d'objets d'intérêt (OI) dans la trace
 - graphe potentiel à instancier
 - contraintes sur la position relative des OI dans l'épisode
 - appartenance à des états, distances
 - contraintes sur la structure interne des OI
 - Annotations
- Explained task signature (EXTASI)

Exemples de signatures de tâches expliquées



Instanciation de signature



Aide / facilitation

- Non pas viser l'appropriation ou bien la bonne utilisation d'un outil
 - relativement à un modèle de conception a priori
- Plutôt fournir à l'utilisateur une ressource pour poursuivre et mener à bien la tâche particulière qu'il a engagée.
- Aide à l'utilisateur plutôt qu'assistance
 - quatre exemples dans la suite



Aide d'un agent assistant spécifique

- Assistant spécifique lié à une tâche / signature spécifique
 - recontextualisation des éléments de l'interaction
- Exemple
 - signaler un utilisateur qu'il a déjà échangé des documents avec le contact qu'il vient d'insérer dans son espace de travail, lors d'une recherche antérieure.
 - basé sur l'Extasi « importer des documents d'un contact »
- Possibilité de RàPC
 - cas = épisode de résolution de problème, reconnu par une signature de tâche
 - assistance à base de cas standard
- Avantage
 - possibilité de partager la même base de cas (trace primitive) entre plusieurs assistants
 - possibilité d'ajouter de nouveaux assistants (nouvelles Extasis) sans changer la trace, en fonction des besoins



Aide d'un agent assistant générique

- Assistant générique
 - moteur d'assistance s'adaptant à une signature de tâche
 - paramétré par une/des signatures
 - exploration de la trace courante et suggestion d'action
 - cf. correcteur orthographique
- Exemple
 - utilisateur en phase de tri de résultats
 - présenter les document non inspectés
 - proposer de supprimer les documents rapidement consultés
- Avantage
 - la tâche est « réifiée » par les signatures, et guide aussi bien l'assistant que l'utilisateur



Aide d'un agent « réactif » par une amorce d'épisode

- Assistant générique lié à plusieurs signatures
 - moteur d'assistance proposant des interprétations de la tâche en cours à l'utilisateur qui le mobilise
 - trouver une signature partiellement instanciée
- Exemple
 - demande d'assistance de l'utilisateur après avoir lancé plusieurs fois une même requête
 - L'assistant reconnaît et propose « Trop de résultats » et « Echec d'une recherche sur le web »
 - L'utilisateur valide la deuxième, l'assistant recherche des épisodes passés correspondant
 - propose d'importer les documents d'un contact
 - propose simplement de les visualiser
- Avantage
 - Les requêtes sont faites simplement en agissant sur le système
 - pas de langage de requête, plus d'à propos



Aide d'un agent pour une tâche non définie

- Assistant générique
 - moteur d'assistance réagissant à une signature de tâche « construite » à la volée par l'utilisateur
- Exemple
 - aucune signature en convient, préférence d'utiliser une base de données en ligne
 - l'utilisateur décrit ce qu'il fait, définissant une nouvelle signature, qu'il apprend au système
- Avantage
 - la tâche est peut-être déjà inscrite dans la trace, mais non connue : la nouvelle Extasi permet alors de proposer de l'aide
 - possibilité de décrire de nouvelles tâches, et de construire sa propre assistance



Applications facilitateurs

- Facilitateurs de tâches liées à l'apprentissage humain (Pixed)
- Facilitateurs de tâches liées à la conception (Dassault)
- Facilitateurs de tâches de veille technologique (Amoweba)
- Facilitateurs de tâches collaboratives (Projets OSCAR, ISOCELE)
- ...



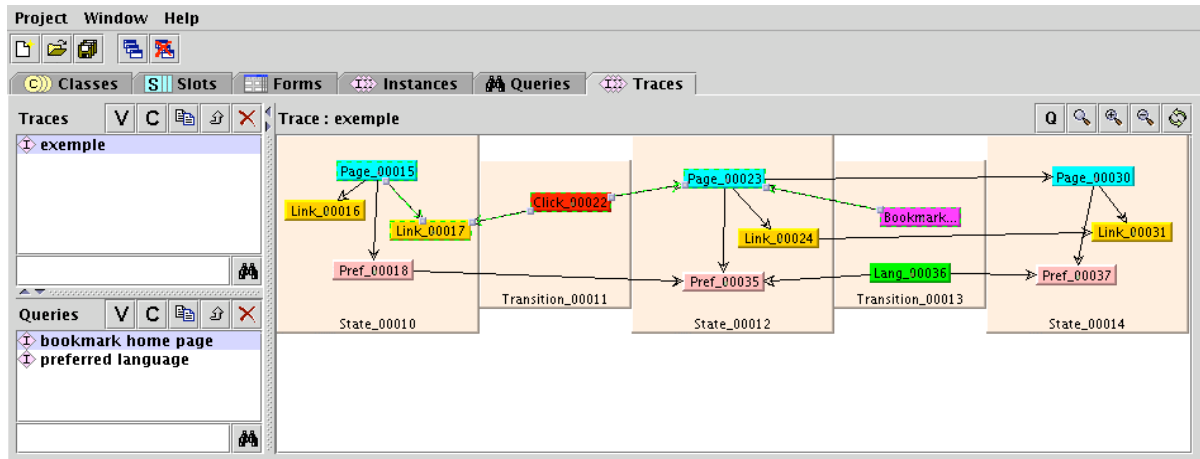
Projets autour de MUSETTE

- Projets fondateurs
 - PIXED
 - RECIS
 - ARDECO
- Projets terminés
 - MAZETTE
 - @CLEVER
 - Thèse Mick Philippon
 - Musette-HL
- Projet en cours
 - Thèse Arnaud Stuber
 - Thèse Olivier Georgeon
 - ACTEURS
 - MNESIS
 - ISOCELE
- Projets au delà de Musette
 - Thèse Julien Laflaquière
 - Thèse Lotfi-Sofiane Settoute
 - Thèse Leila Yahiaoui
 - ...



Atelier Musette / Protege

- Pour concevoir des modèles d'utilisation



Plan

- Introduction : pourquoi les traces ?
- Approche Musette
- **Systèmes à base de traces**
- Exemples de travaux
- Conclusion



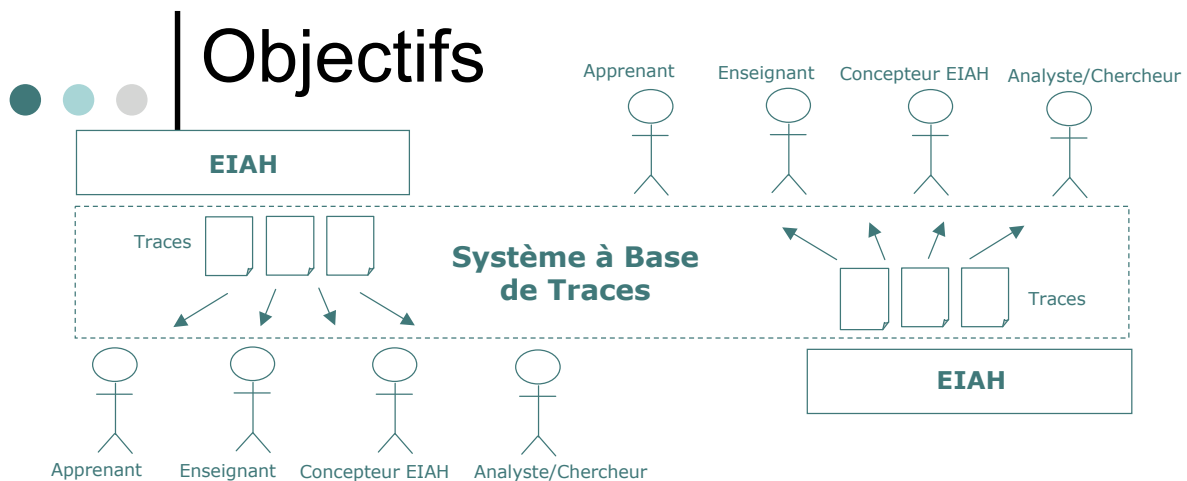
Pourquoi les SBT ?

- Généralisation approche Musette
 - Unification modèles / outils
 - Principe général
 - modélisation explicite de la trace pour attaquer la complexité du système visé
- Cadre Cluster Région EIAH
 - Thèse Lotfi-Sofiane Settouti
 - Contexte EIAH
- Présentation orientée EIAH, concepts valides pour n'importe quel domaine



Traces dans les EIAH

- Outil pour le chercheur
 - analyse des situations d'apprentissage, analyse des usages et des comportements des utilisateurs des EIAH, profils d'apprenants...
- Outil pour l'enseignant
 - régulation des situations d'apprentissage
- Outil pour l'apprenant
 - réutilisation des trace, visualisation de la trace
- Outil pour le concepteur
 - amélioration de la conception et des interfaces...



- **Systèmes à base de traces**
 - systèmes informatiques facilitant la manipulation des traces
- **Nécessité d'un cadre méthodologique et théorique**
 - Méthodologie pour la mise en place d'études à base de traces.
 - Cadre commun et valide permettant d'échanger et de parler des traces, leurs traitements et leurs exploitations.

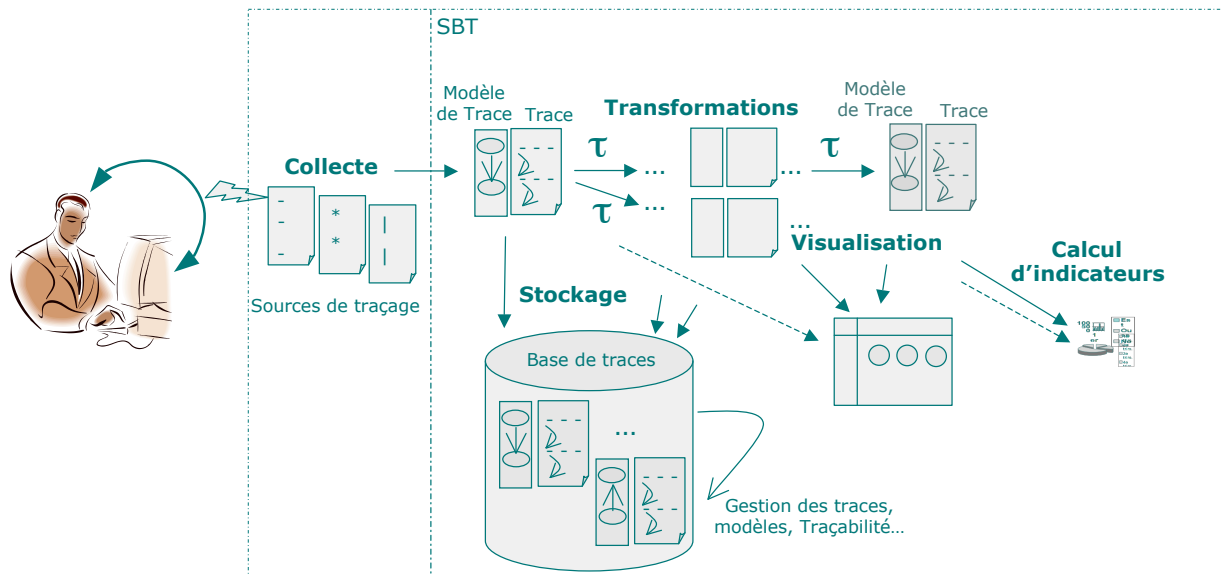
Notion de trace et de SBT

- **Définition**
 - Trace informatique = séquence temporelle d'observés
 - Ensemble de données temporellement situées
 - Les observés sont des données relatives à une observation faite d'une activité
- **Système à Base de trace**
 - Système informatique permettant l'exploitation des traces



Systemes à base de traces

- o Gestion du cycle de vie des traces

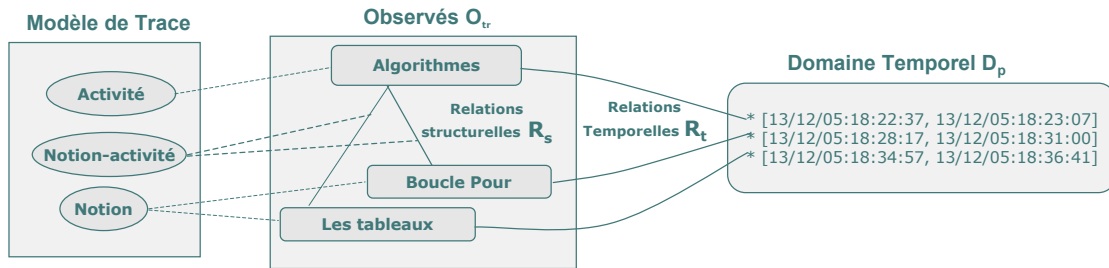


Formalisation des traces

- o Une trace est un 4-uplet $\text{Trace}=(D_p, O_{tr}, R_t, R_s)$ où
 - D_p un domaine temporel. $D_p = (T, \leq)$ avec
 - T ensemble fini d'intervalles
 - \leq une relation d'ordre dans T .
 - Les intervalles de temps sont décrits par T_d et T_f et notés $[T_d, T_f]$.
 - O_{tr} ensemble fini des observés de la trace, $O_{tr} = \{o_1, o_2, \dots, o_n\}$
 - R_t une relation représentant les liens temporels $D_t \times O_{tr}$
 - tel que $R_t \subseteq D_t \times O_{tr}$
 - R_s une relation représentant les liens structurels $O_{tr} \times O_{tr}$
 - tel que $R_s \subseteq O_{tr} \times O_{tr}$
- o Un modèle de trace est une structure $\Theta=(\Theta_c, \Theta_r)$ où
 - $\Theta_c = \{\theta_{c1}, \theta_{c2}, \dots, \theta_{cn}\}$
 - $\Theta_r = \{\theta_{r1}, \theta_{r2}, \dots, \theta_{rn}\}$

Exemple

- Modèle de la trace Pixed
 - $\Theta_c = \{\text{Activité, notion}\}$
 - $\Theta_r = \{\text{Notion-Activité}\}$



Représentation d'une trace du système Pixed

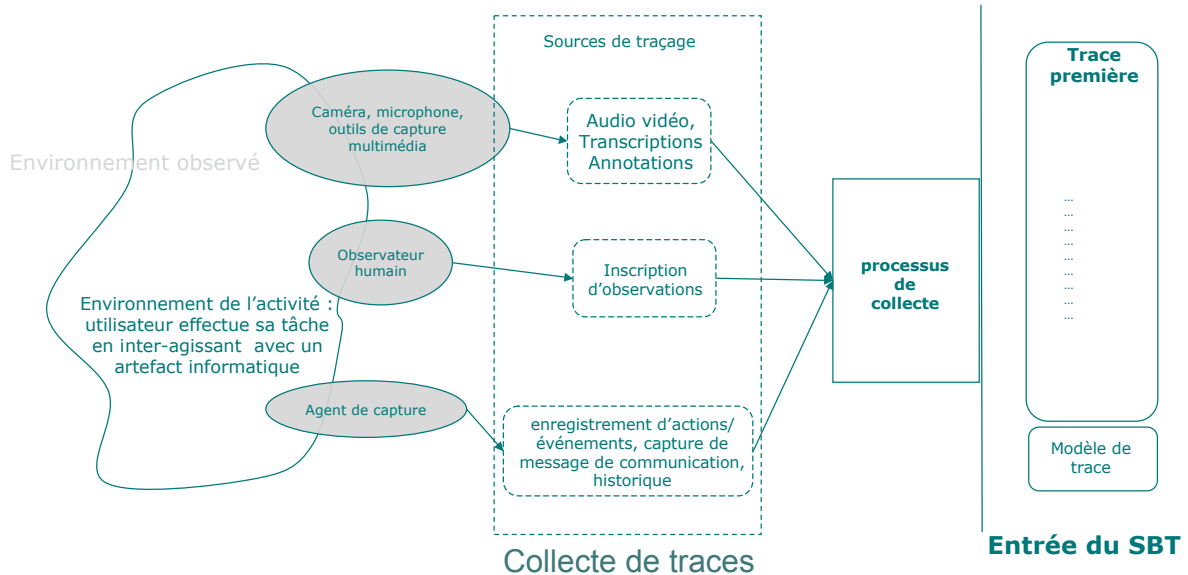
Services du SBT

- Collecte de traces
 - Transformations des traces
 - Gestion des traces
 - Stockage de traces
 - Visualisation des traces
- SBT Kernel
- Système à Base de Traces



Collecte de traces

- Ensemble structuré de processus permettant de convertir une ou plusieurs sources de données en une trace première



Définitions

- Source de traçage
 - fichier ou flux de données dans un format explicite quelconque (ex : vidéo , fichiers log...)
- Trace première
 - Trace SBT construite par un processus de collecte. Trace non transformée.
- Trace SBT
 - Séquence Temporelle d'observés associée à un Modèle de trace explicite représentant le vocabulaire de la trace et sa sémantique
- Base de traces
 - ensemble des traces du SBT et de leurs modèles. Elle permet le stockage permanent et l'accès à tout moment aux traces Modèle de trace
 - les traces d'entrée sont des traces primitives. Les autres sont des traces transformées



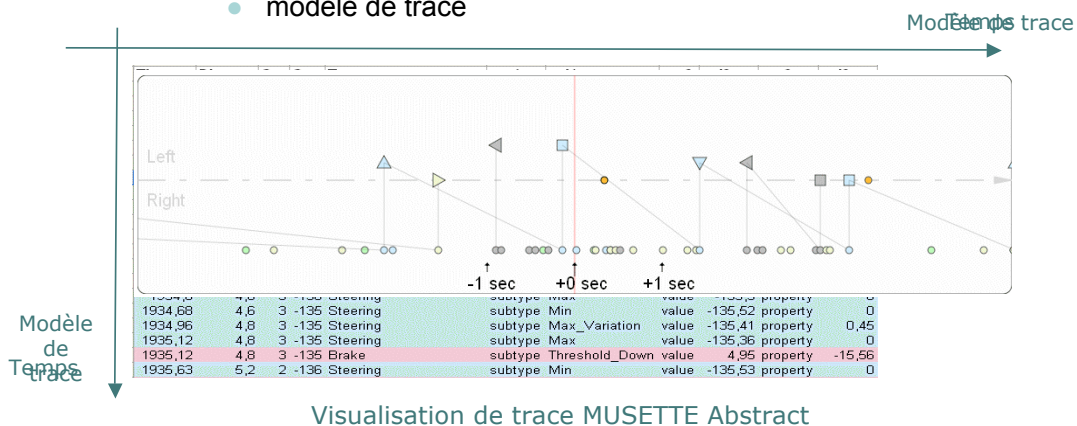
Gestion de traces

- Gestion des éléments de la base de trace
 - traces
 - modèles
 - modèles de trace / modèles de transformation
 - requêtes
- Fonctionnalités
 - mises à jour de la base de traces
 - conservation des traces
 - droits d'accès sur les traces
- Gestion de l'évolution des traces par transformation
 - assurer la traçabilité des transformations effectuées sur les traces



Visualisation de trace

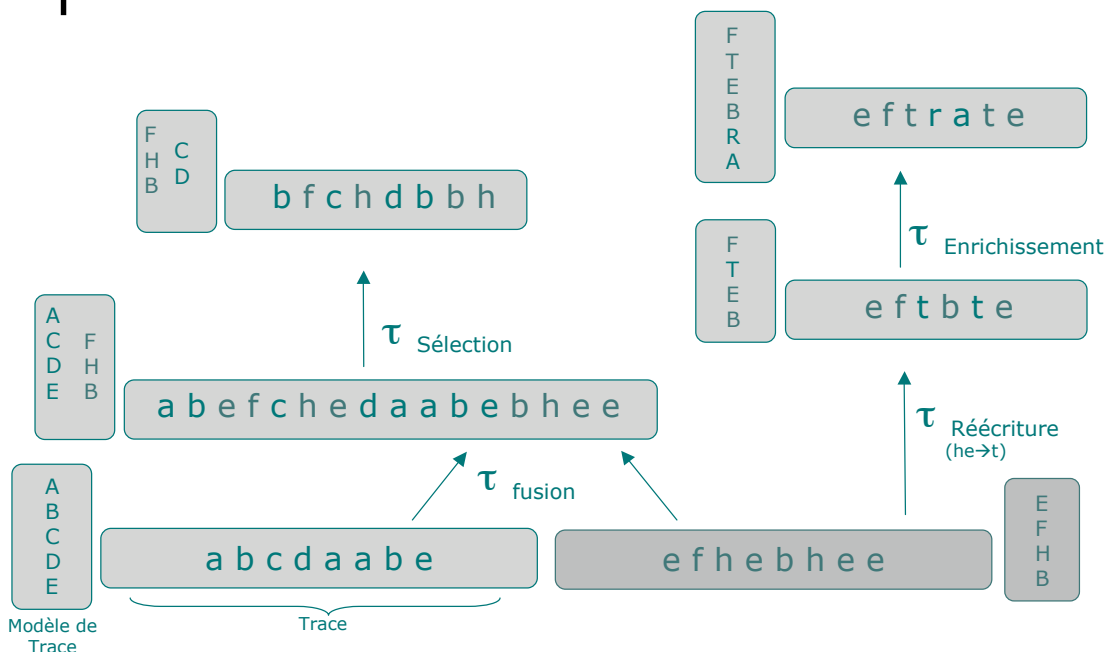
- Système de visualisation : ensemble des techniques de présentation des traces
 - Une visualisation selon une ligne de temps permettant de représenter la séquence temporelle de données
- Exemple : visualisation selon deux dimensions
 - axe temporel
 - modèle de trace



● ● ● | Système d'interrogation

- Interroger les traces suivant plusieurs dimensions
 - Modèle de trace
 - Exemple : Une trace ayant le modèle (apprenant, activité_pédagogique, notion)
 - Requête : donner les observés ayant comme activité pédagogique « Algorithmie »
 - Domaine temporel
 - Donner les observés dans l'intervalle [10h:30:22,10h:40:44]
- Interroger la base de traces
 - Ex . Donner les traces relatives à l'expérimentation « X » et relatant les notions « Y »
 - Donner les traces transformée par réécriture entre le 29/04/2006 et le 10/05/2006

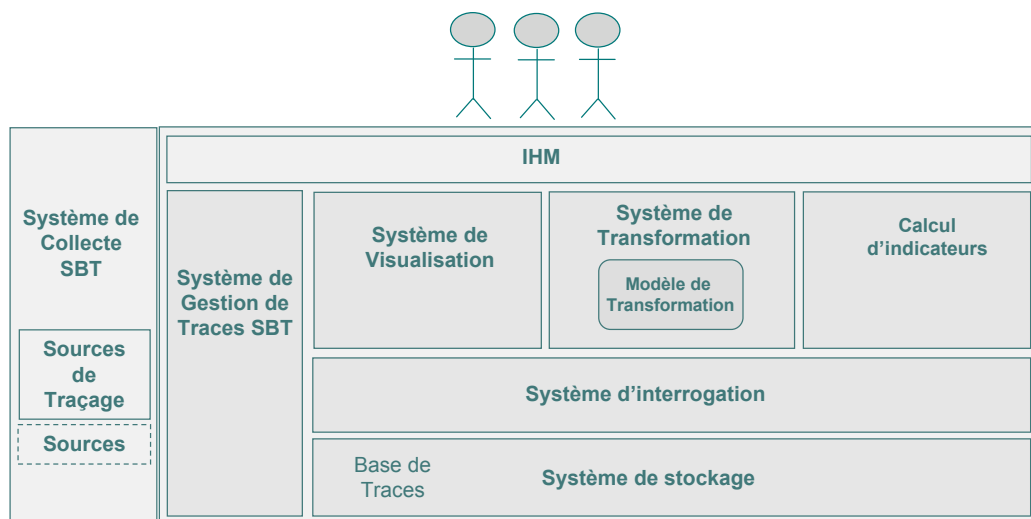
● ● ● | Transformations



● ● ● | Système de Transformation

- Système permettant d'effectuer des transformations
- Un transformation τ
 - s'opère sur une ou plusieurs traces en entrée et a pour résultat une nouvelle trace appelée trace transformée
 - peut être opérée
 - manuellement
 - automatiquement, à partir d'un modèle de transformation
 - ensemble de règles formelles explicitées dans un certain langage
- Quelques transformations possibles
 - enrichissement de la trace
 - sélection
 - fusion selon les domaines temporels
 - agrégation par réécriture de motifs
 - ...

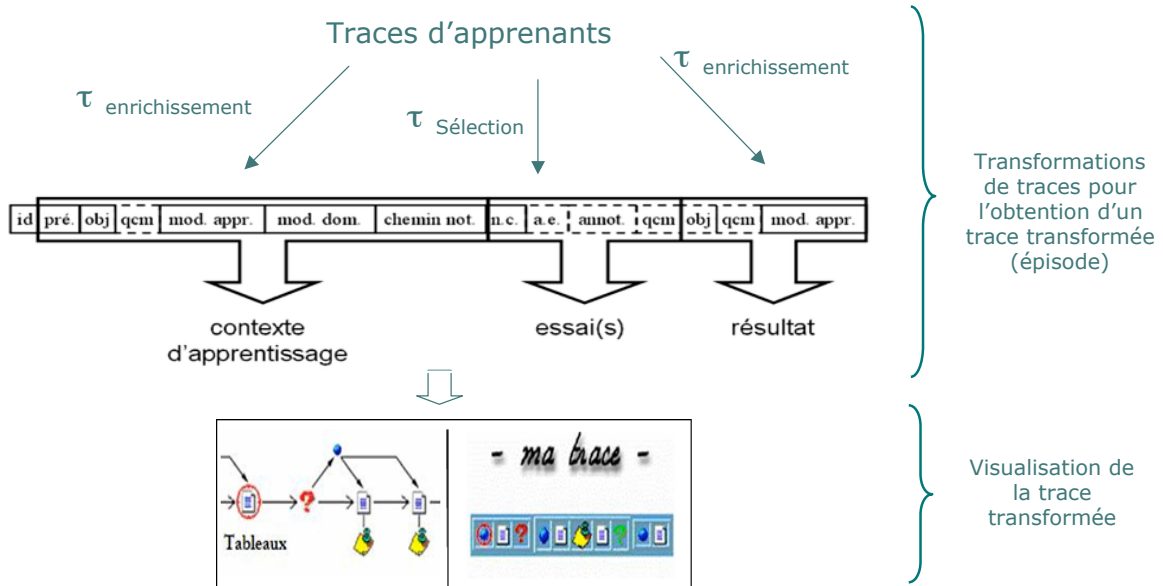
● ● ● | Système à base de traces (SBT)



Architecture du système à base de traces

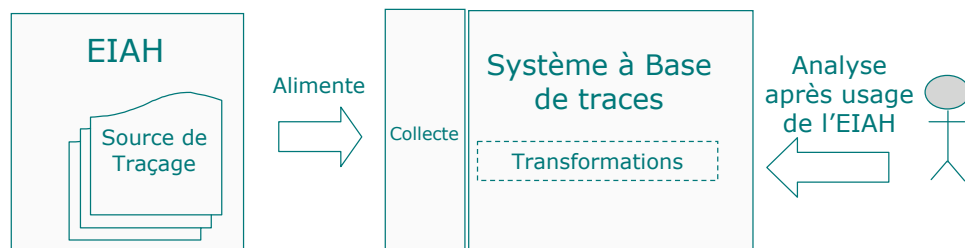


Exemple de transformation Pixed

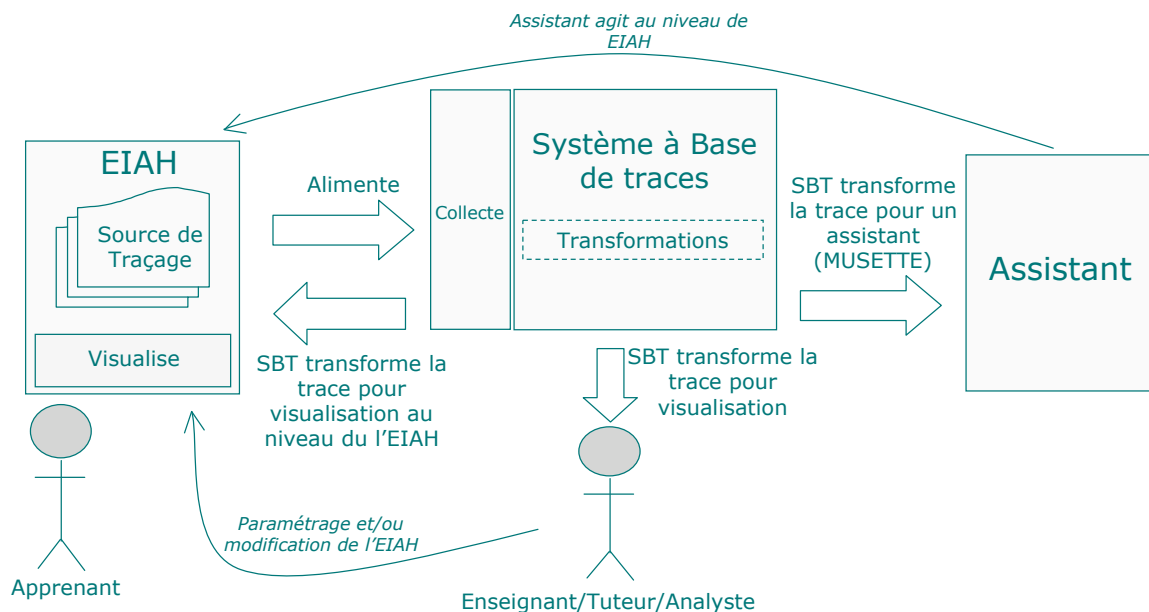


Framework SBT : analyse

- Construction de Systèmes à Base de Trace pour permettre de comprendre l'activité et le comportement des utilisateurs d'un EIAH



Framework SBT : facilitation



SBT : conclusion

- Cadre théorique et méthodologique pour l'exploitation des traces
 - Notion de « système à base de traces » et ses services
 - Architecture des SBT
 - Modélisation formelle :
 - traces+modèle de traces
 - transformations de traces
- Développements
 - Framework informatique SBT



SBT : suites

- Niveau théorique
 - Composition de transformations
 - Visualisations
- Niveau pratique
 - Construction framework
 - Construction SBT réels




Plan

- Introduction : pourquoi les traces ?
- Approche Musette
- Systèmes à base de traces
- **Exemples de travaux**
- Conclusion



Exemples

- Présentation de la trace : PIXED
- Analyse : ABSTRACT
- Analyse, présentation temps réel : eLycee



Facilitateur : exemple Pixed

- PIXED
 - Projet d'Intégration de l'Expérience pour l'Enseignement à Distance (thèse Jean-Mathias Héraud - 2002)
- Réseau de notions (relations didactiques)
- Activités éducatives attachées aux notions
- Annotations nombreuses et variées par l'apprenant / l'enseignant
- Facilitation de l'orientation dans le processus d'apprentissage : approche « singulière »

Annoter dans Pixed

- enseignants -

- Vision -

[retour](#)

[éditer le modèle](#)

[augmenter le contenu](#)

[annoter le contenu](#)

[administration](#)

- annoter -

novice intermédiaire expert

Le capteur à CCD (charge-coupled device), ou capteur à transfert de charge est un composant basé sur une technologie de composants à semi-conducteurs. L'arrivée de photons sur le silicium va créer des charges électriques, charges qui vont s'accumuler dans des zones appelées "puits". Ces zones sont la résultante de différentes tensions électriques appliquées localement. En jouant sur ces tensions, nous allons modifier la position des puits, et donc forcer les charges à se déplacer.

Transfert de trame/Interligne :

Une matrice CCD est formée de Y lignes de X colonnes, chaque pont élémentaire (ou pixel) étant physiquement un rectangle de silicium de quelques microns. Il va falloir déplacer les charges accumulées dans ce pixel avant de les convertir en tension électrique, déplacement qui peut atteindre une quinzaine de millimètres, ce qui n'est pas instantané. Il faut donc disposer d'une zone de stockage intermédiaire pour transférer des données. C'est sur la définition de cette zone de stockage que deux technologies s'opposent :

a) **Le transfert de trame :**
 A la fin du temps d'intégration, les charges équivalentes mais protégées de la lumière par un puits par pixel pour donner le signal vidéo.

Copier
 Copier
 Coller
 Sélectionner tout
 Imprimer
 Annoter
 Expert
 Moyen
 Novice

Modélisation et utilisation

Annoter dans Pixed

- étudiants -

- Vision Artificielle -

[retour](#)

navigation linéaire

Capteur à CCD :

Principe :

Nouvelle annotation - Pixed -

Titre : Capteur à CCD

Commentaire : Point ou pont ?

à destination de : enseignants

Auteur : heraud@univ-savoie.fr

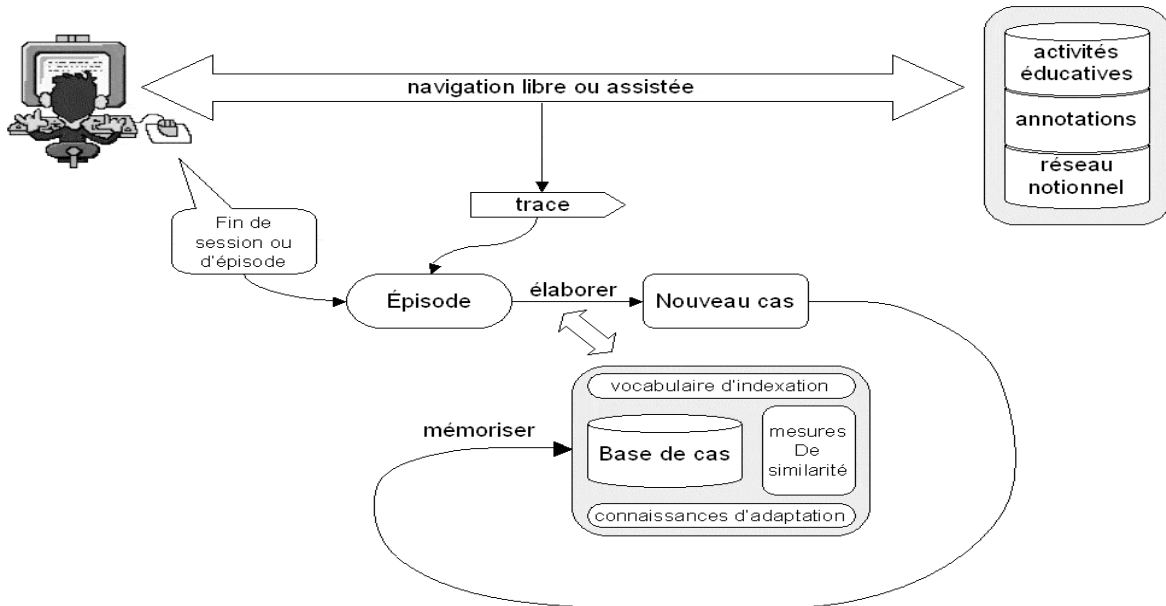
Annuler ! Annuler

lumière → CCD → capteur → acquisition

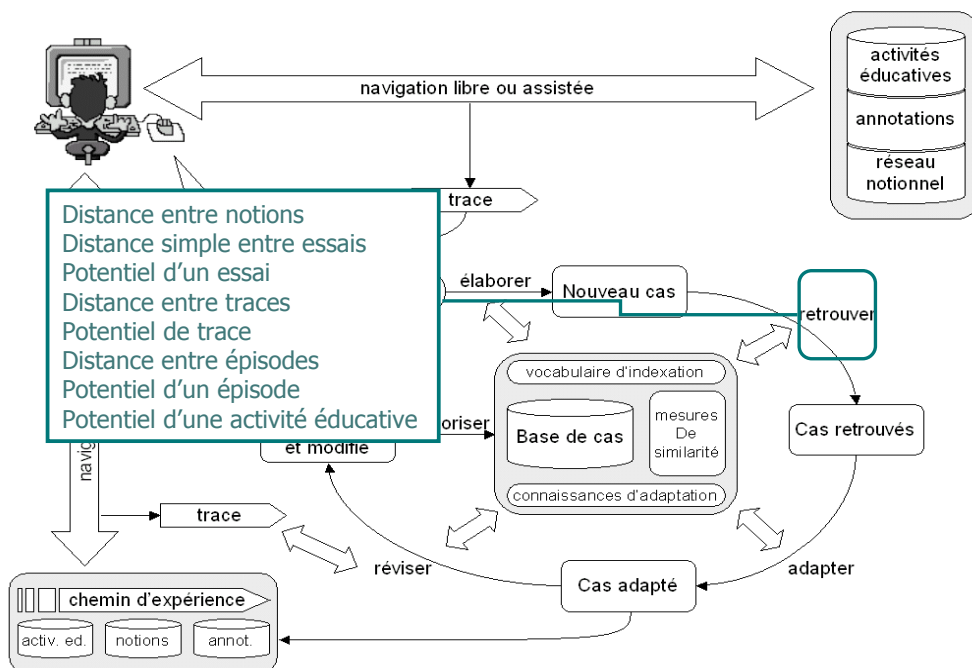
Une matrice CCD est formée de Y lignes de X colonnes, chaque pont élémentaire (ou pixel) étant physiquement un rectangle de silicium de quelques microns. Il va falloir déplacer les charges accumulées dans ce pixel avant de les convertir en tension électrique, déplacement qui peut atteindre une quinzaine de millimètres, ce qui n'est pas instantané. Il faut donc disposer d'une zone de stockage intermédiaire pour découpler le temps d'intégration du temps de transfert des données. C'est sur la définition de cette zone de stockage que deux technologies s'opposent :

Modélisation et utilisation

Amorçage de la base d'épisodes



Cycle de réutilisation de l'expérience tracée





Ma trace

Accueil étudiant - LISA - Internet Explorer
http://welcome.to/pixed/

Pixed
- étudiants -

- Les salaires -
[retour](#) [aide](#) [trace](#)

Les enregistrements comptables des contributions sociales

Salaire brut
Définition : Le salaire brut est la rémunération globale du travail. C'est un élément définitif à la charge de l'entreprise.

Informations à enregistrer
Les informations qui contribuent au calcul du salaire brut sont les suivantes :

- **les contributions sociales**
elles sont déduites du salaire, elles font l'objet de versements de la part de l'employeur aux organismes concernés
- **le salaire net**
c'est le salaire réellement payé au salarié

Illustration
Cette illustration présente clairement les éléments constitutifs du salaire brut et leur répartition.

Situations intermédiaires

Eléments définitifs
Salaires bruts - **Cotisations sociales** = **Salaires nets à percevoir**

Diagramme de flux : livre de paie → enreg. compt. → mod. feuilles

Terminé

Modélisation et utilisation de



Episode / Trace

Pixed
- étudiants -

- Vision Artificielle -
[retour](#) [aide](#) [???](#)

Capteur à CCD :

Principe :

Le capteur à CCD (charge-coupled device), ou capteur à transfert de charge est un composant basé sur une technologie de composants à semi-conducteurs. L'arrivée de photons sur le silicium va créer des charges électriques, charges qui vont s'accumuler dans des zones appelées "puits". Ces zones sont la résultante de différentes tensions électriques appliquées localement. En jouant sur ces tensions, nous allons modifier la position des puits, et donc forcer les charges à se déplacer.

Transfert de trame/Interligne :

Une matrice CCD est formée de Y lignes de X colonnes, chaque pont élémentaire (ou pixel) étant physiquement un rectangle de silicium de quelques microns. Il va falloir déplacer les charges accumulées dans ce pixel avant de les convertir en tension électrique, déplacement qui peut atteindre une quinzaine de millimètres, ce qui n'est pas instantané. Il faut donc disposer d'une zone de stockage intermédiaire pour découpler le temps d'intégration du temps de transfert des données. C'est sur la définition de cette zone de stockage que deux technologies s'opposent :

Diagramme de flux : Capteur à CCD → ? → [] → []

Modélisation et utilisati



ABSTRACT

- Thèse O. Georgeon



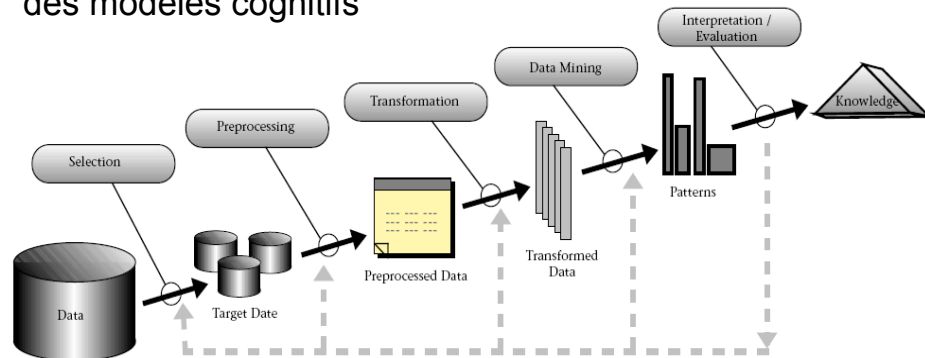
ESDA

- Exploratory sequential data analysis
 - Any empirical undertaking seeking to analyze systems, environmental, and/or behavioral data (usually recorded) in which the sequential integrity of events has been preserved.
- Données
 - Enregistrements dans une voiture instrumentée = Traces
- Objectifs ESDA
 - Découvrir des connaissances par analyse d'enregistrements séquentiels.
 - Réduire le rapport temps d'analyse / temps réel de l'enregistrement
- Exploration par opposition à confirmation:
 - On se focalise sur la découverte de connaissances, et non sur l'analyse statistique à but de confirmation d'hypothèse préétablies.
 - On cherche à "looking at data to see what it seems to say"
 - La méthode itérative et non univoque
- Importance des transformations de données
 - Transformation de traces dans un SBT



Analyse de trace comme ECD

- Données : trace = données séquentielles
- Utilisateur : psychologue
- Connaissance : schémas de conduite, prédictions de comportement
- But : aider le psychologue à trouver des modèles cognitifs

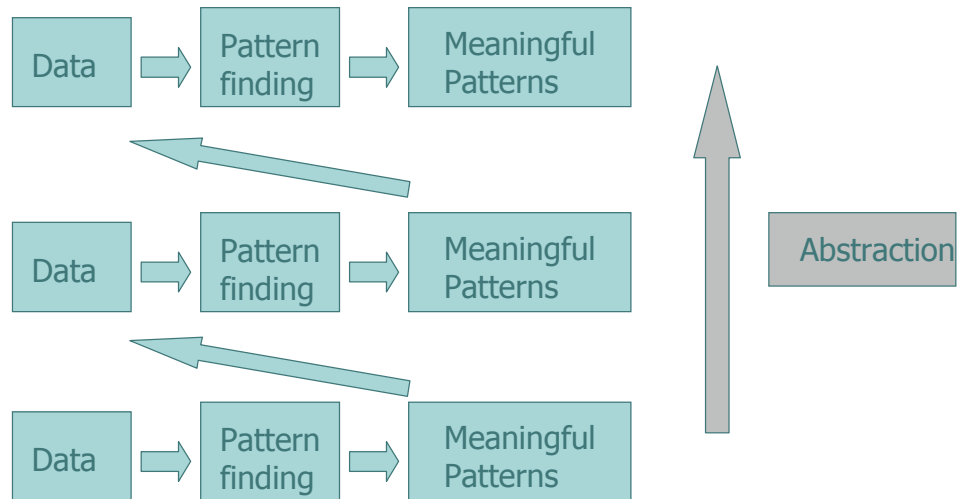


Proposition

- Construction interactive du sens
 - Ne vient pas du système qui ne comprend rien aux données
 - Vient de l'utilisateur qui utilise interactivement le système
- L'expertise de l'analyste peut être modélisée dans le système
 - Fait partie de la connaissance découverte
 - Exemple
 - Règles de transformation / inférences



Abstraction et recherche de patterns



Niveaux d'abstraction

- Niveau 0
 - Sources de traçage et collecte
- Niveau 1
 - Trace première
 - Données nettoyées
- Niveau 2
 - Trace transformée
- Niveau 3
 - Reconnaissance de schéma
 - Ex. Changement de voie

Sources de traçage et collecte de la trace



Premier niveau d'abstraction

5876	3806.32	18496	112	5	Speed	subtype	Min	value	111.81	property	0
5877	3807.21	18523.4	112	3	Accelerator	subtype	Max	value	50.35	property	0
5878	3807.71	18539.6	113	4	Eye_Left	duration		0.0499985			
5879	3807.76	18540.8	113	4	Eye_Far_Left	duration		0.149996			
5880	3807.77	18540.8	113	4	Speed	subtype	Max_Variation	value	112.59	property	0.81
5881	3807.91	18545.8	113	3	Eye_Left_Mirror	duration		0.116997			
5882	3808.03	18549	113	3	Eye_Far_Left	duration		0.0169995			
5883	3808.04	18549.6	113	3	Eye_Left	duration		0.066998			
5884	3808.05	18549.6	113	3	Speed	subtype	Max_Variation	value	112.79	property	0.69
5885	3808.11	18551.4	113	3	Eye_Ahead	duration		2.40493			
5886	3808.77	18572.2	113	5	Accelerator	subtype	Min	value	46.98	property	0
5887	3808.81	18573.4	113	5	Speed	subtype	Max_Variation	value	113.33	property	0.76
5888	3808.93				Close_Left_Up						
5889	3808.93	18577.2	113	5	Steering	subtype	Threshold_Up	value	5.03	property	1.13
5890	3808.99	18579.2	113	5	Speed	subtype	Max_Variation	value	113.44	property	0.68
5891	3809.03	18580.4	114	5	Accelerator	subtype	Max	value	46.99	property	0
5892	3809.38	18591.8	114	6	Accelerator	subtype	Min	value	46.92	property	0



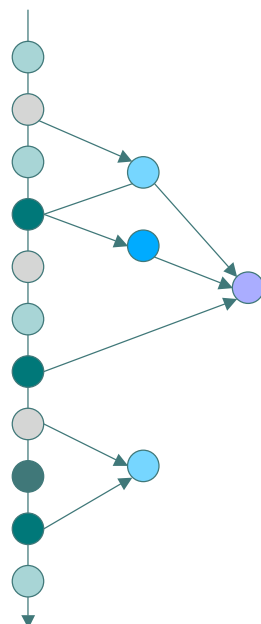
Du premier au second niveau

5876	3806.32	18496	112	5	Speed	subtype	Min	value	111.81	property	0
5877	3807.21	18523.4	112	3	Accelerator	subtype	Max	value	50.35	property	0
5878	3807.71	18539.6	113	4	Eye_Left	duration		0.0499985			
5879	3807.76	18540.8	113	4	Eye_Far_Left	duration		0.149996			
5880	3807.77	18540.8	113	4	Speed	subtype	Max_Variation	value	112.59	property	0.81
5881	3807.91	18545.8	113	3	Eye_Left_Mirror	duration		0.116997			
5882	3808.03	18549	113	3	Eye_Far_Left	duration		0.0169995			
5883	3808.04	18549.6	113	3	Eye_Left	duration		0.066998			
5884	3808.05	18549.6	113	3	Speed	subtype	Max_Variation	value	112.79	property	0.69
5885	3808.11	18551.4	113	3	Eye_Ahead	duration		2.40493			
5886	3808.77	18572.2	113	5	Accelerator	subtype	Min	value	46.98	property	0
5887	3808.81	18573.4	113	5	Speed	subtype	Max_Variation	value	113.33	property	0.76
5888	3808.93				Close_Left_Up						
5889	3808.93	18577.2	113	5	Steering	subtype	Threshold_Up	value	5.03	property	1.13
5890	3808.99	18579.2	113	5	Speed	subtype	Max_Variation	value	113.44	property	0.68
5891	3809.03	18580.4	114	5	Accelerator	subtype	Max	value	46.99	property	0
5892	3809.38	18591.8	114	6	Accelerator	subtype	Min	value	46.92	property	0

- Eye_sequence_end: Eye_Ahead plus de 0.9s
- Short_Left_Mirror_Glance: Sequence de moins de 0.8s incluant au moins un Eye_Left_Mirror



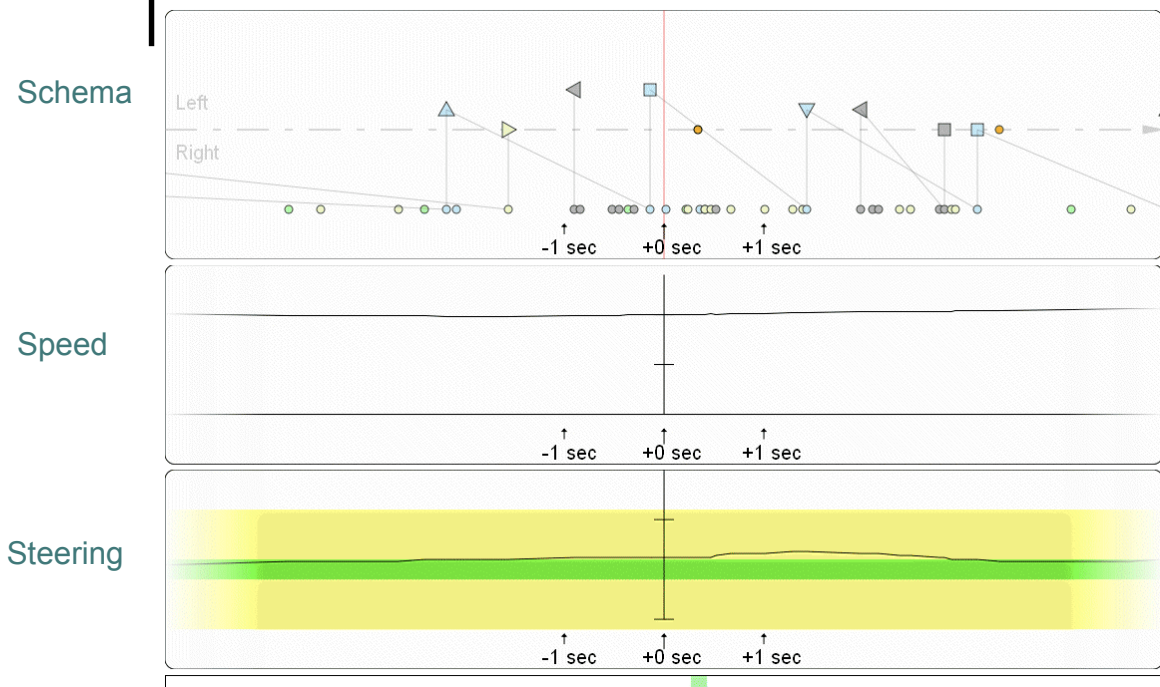
Règles de transformation pour second niveau d'abstraction



- La trace est un graphe (RDF)
- Les événements sont définis par l'utilisateur dans des ontologies (RDFS)
- Les transformations / inférences sont définies par l'utilisateur comme des requêtes (SPARQL)



Présentation second niveau d'abstraction



Explications

- Couleurs :
 - bleu: volant
 - ocre: vitesse
 - vert: accumulateur
 - rose: frein
 - orange: clignotant
 - gris: regards
- Axe en pointillé mixte représente la direction longitudinale
 - objets dessus --> régulation longitudinale
 - au dessus --> régulation latérale à gauche
 - au dessous --> régulation latérale à droite.
- Les carrés représentent l'idée de stabilité, et les triangles l'idée de changement.
- Exemples
 - Un carré bleu au dessus de l'axe indique que le volant est stabilisé en "tourne à gauche", un carré bleu sur l'axe indique que le volant est stabilisé tout droit. Un carré gris sur l'axe indique un début de regard tout droit. Un triangle ocre sur l'axe pointé vers l'avant indique que la vitesse augmente, pointé vers l'arrière: que la vitesse diminue. Un triangle gris pointé vers l'arrière au dessus de l'axe indique une séquence de regard dans le rétroviseur gauche. pointé vers le haut au dessus de l'axe: un regard à gauche. Un triangle bleu au dessus de l'axe pointé vers le haut indique un début de tourne à gauche. Un triangle bleu au dessus de l'axe pointé vers le bas indique le début de ramener le volant tout droit.



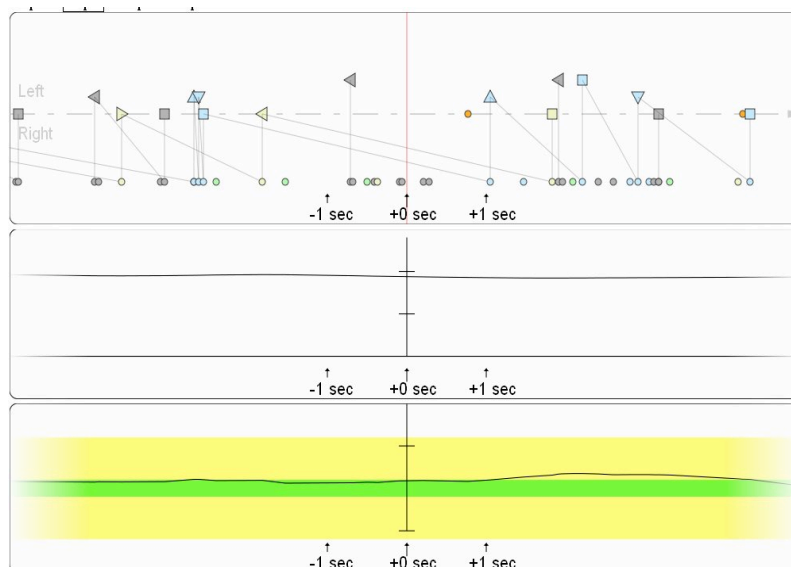
Changement de voie : vers le troisième niveau d'abstraction



- Objectif : reconnaître une configuration indiquant un changement de voie

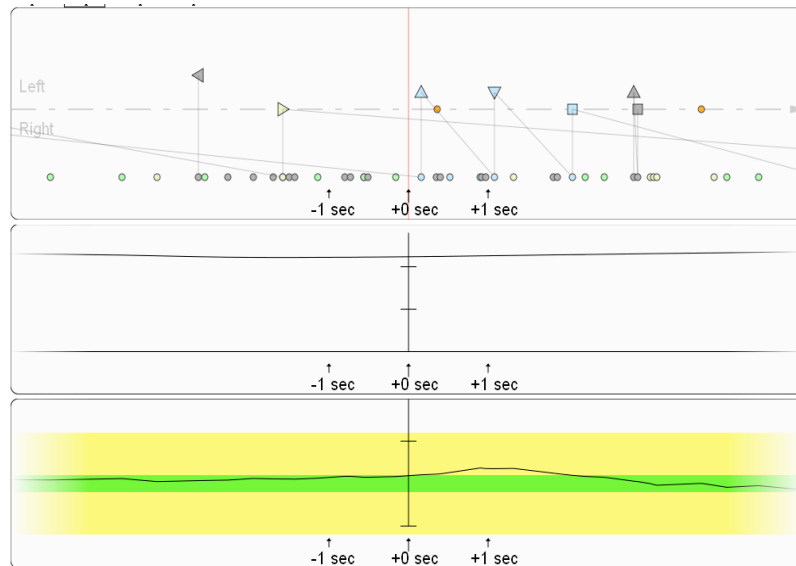


Changement de voie (2)

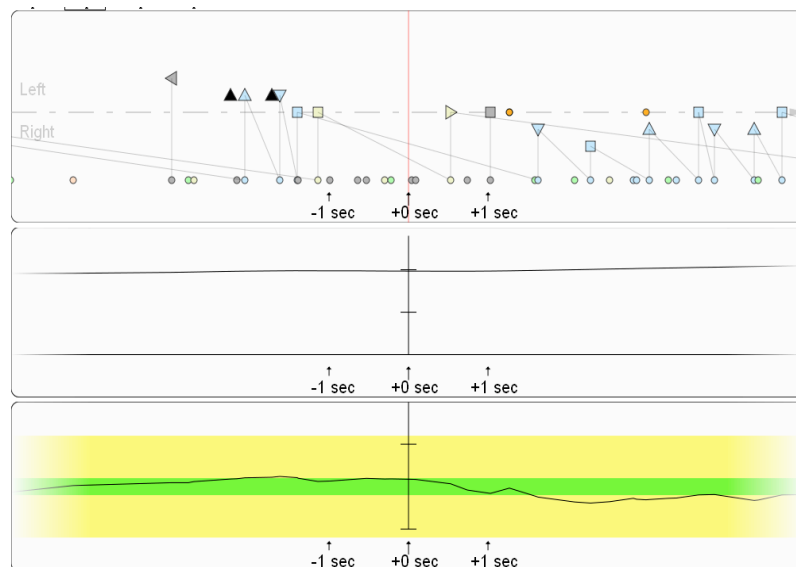




Changement de voie (3)

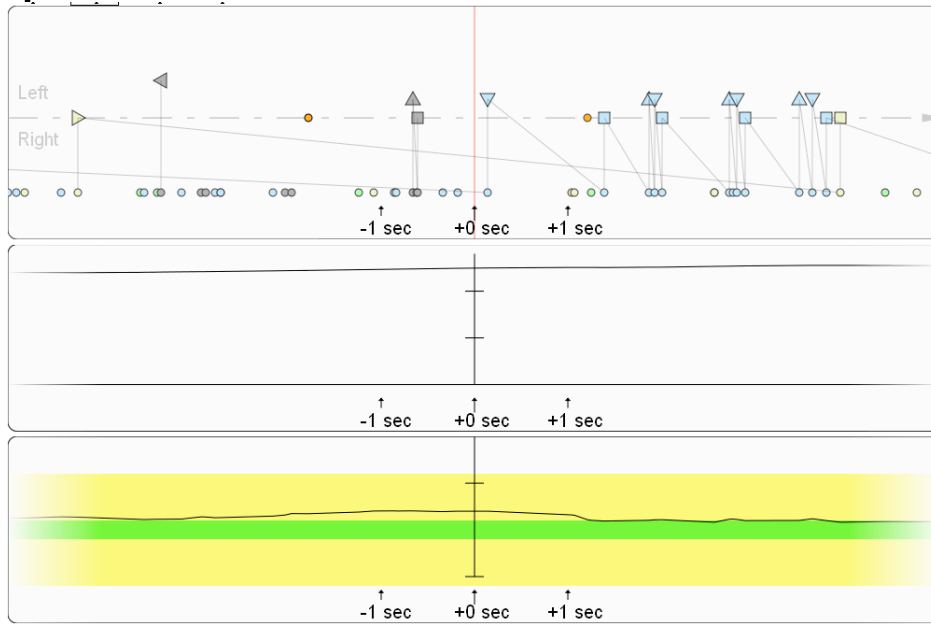


Changement de voie (4)

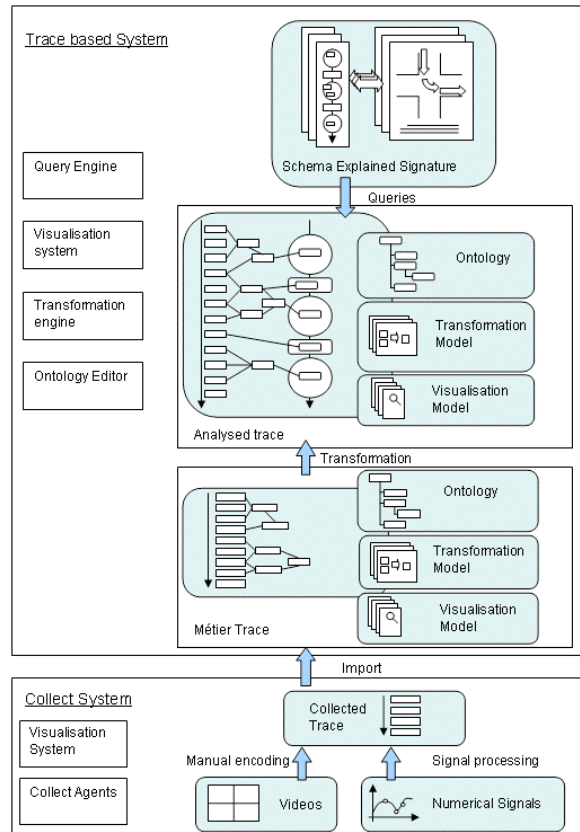




Changement de voie (5)



Architecture





Projet eLycée

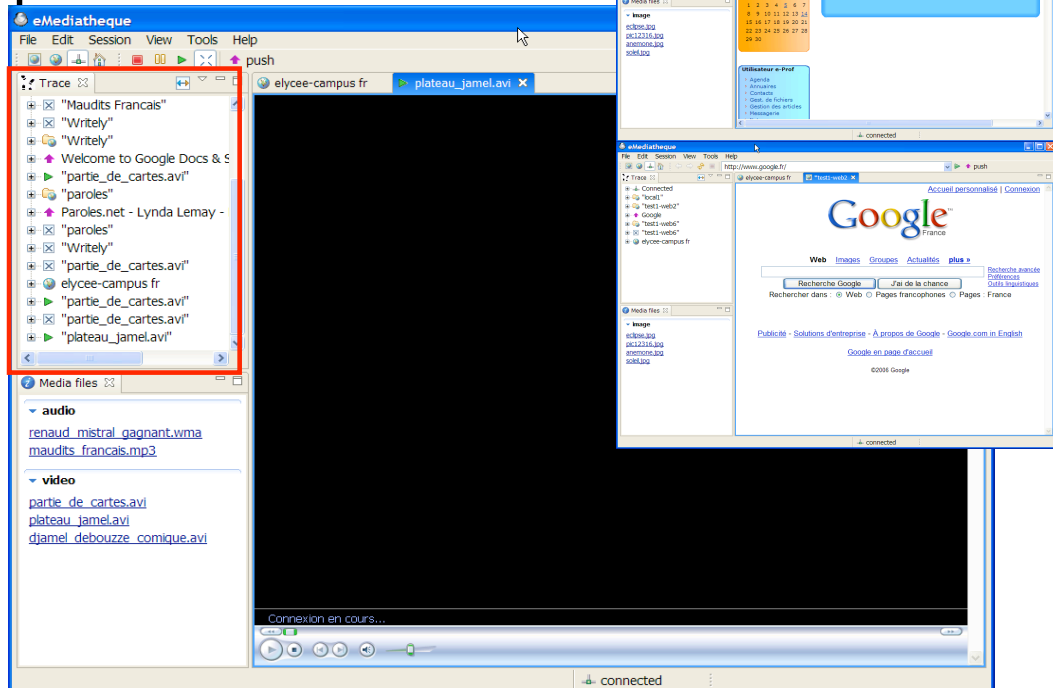
- Collaboration LIRIS (Silex) / eLycée S.A.S.
 - volet développement
 - eMediatheque
 - volet recherche
 - étude de l'influence réflexive des traces sur l'apprentissage et l'appropriation de l'outil
- Activité de eLycée
 - enseignement à distance de la langue et de la culture française en classes virtuelles interactives
 - Utilisation massive de rich media (clips, etc.)
 - Co-construction (par classe, par sous-groupe)
- Marché prioritaire: lycéens français aux USA
 - env. 50% n'ont pas de scolarité française



Classes virtuelles

- Plateforme collaborative eMediathèque
 - Co-navigation Web (= navigation partagée)
 - Co-visualisation de fichiers média (vidéo, audio, images)
 - Interface entièrement personnalisable permettant la manipulation/navigation entre plusieurs documents
 - Intégration de services
 - Ovidentia, Marratech, Wildfire, Writely, Wordpress, et probablement éduéo (en négociation)
 - Fonctionnalités futures
 - chat (instant messaging)
 - Whiteboard
- Marratech
 - Vidéo/audio conférence multi parties
 - les élèves voient le prof et les autres élèves aussi
 - Possibilité de former des binômes
 - audio et *chat* privés
 - Tableaux blancs, multiples et partagés

eMediatheque



Modélisation et utilisation de traces d'expérience d'utilisation de systèmes informatiques – M2R 06/07 – Yannick Prié – LIRIS

85

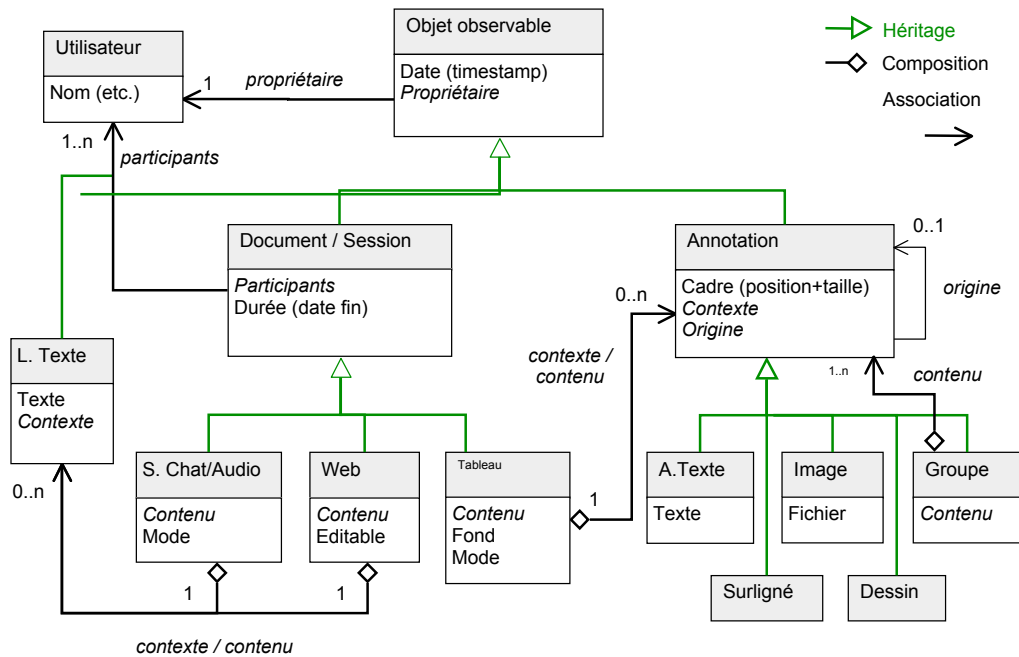
Premiers tests eLycée

- Premières classes de test avec eMediatheque
 - en septembre et octobre
 - environ 12 lycéens français, 2 enseignants
 - observation par un spécialiste CSCW
- Retours
 - Déjà une certaine efficacité
 - Trace souvent perçue comme vecteur privilégié de fonctionnalités perçues comme manquantes et demandées
- Recherche qui débute
 - Traces pour la réflexivité
 - Traces pour l'analyse
 - collaboration, apprentissage
 - Visualisation de traces
 - Plusieurs membres de Silex impliqués

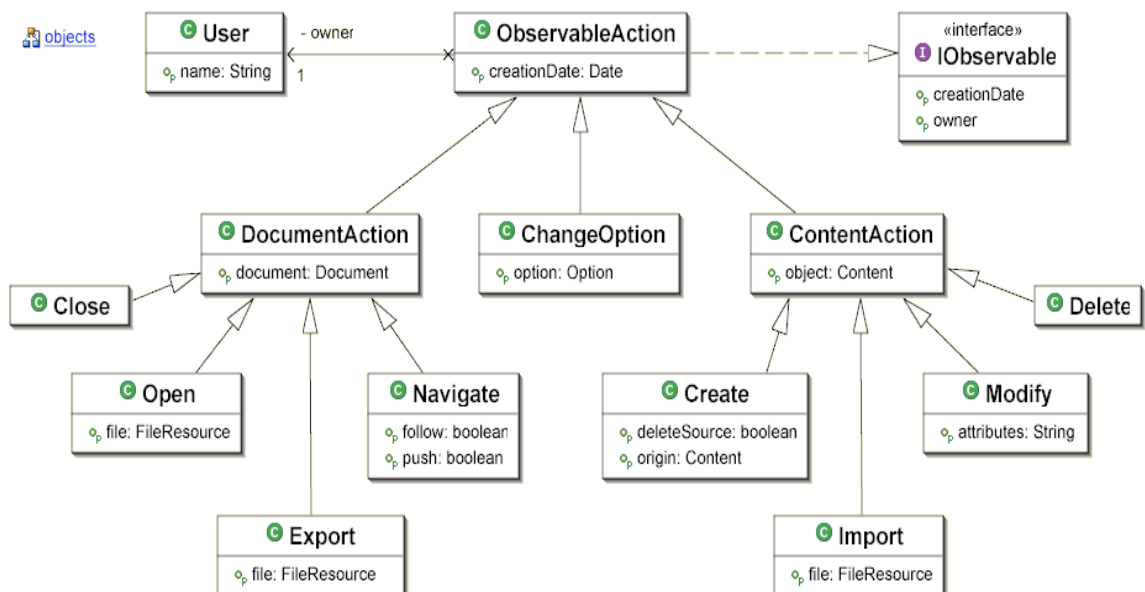
Modélisation et utilisation de traces d'expérience d'utilisation de systèmes informatiques – M2R 06/07 – Yannick Prié – LIRIS

86

Modèle des objets observables



Modèle des actions





Autres travaux

- Musette analyse + facilitation apprentissage humain
 - GeoNotes
- SET (Systèmes d'Exploitation intégrant des Traces)
 - les traces comme éléments « naturels » des systèmes d'exploitation
- Ambre-Musette
 - Utilisation de traces dans un système à base de connaissances lié à un EIAH
- ...



Compléments au cours

- A consulter
 - Article + transparents Rochebrune 2006
 - la trace comme arène résolution du hiatus entre conception et utilisation
 - Article + transparents KES-EME 2006
 - la trace comme outil réflexif pour les espaces documentaires complexes



Plan

- Introduction : pourquoi les traces ?
- Approche Musette
- Systèmes à base de traces
- Exemples de travaux
- **Conclusion**



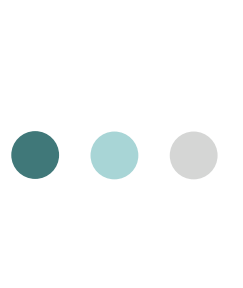
Conclusion

- Trace et facilitation
- Systèmes à Base de Traces
- Travaux en cours
 - visualisation de traces
 - documentarisation de traces
 - ...



Remerciements

- Membres de l'équipe Silex qui travaillent sur les traces
- Plus particulièrement
 - Denis Jouvin
 - Olivier Georgeon
 - Julien Laflaquière
 - Lotfi-Sofiane Settouti



Annexes

Quelques problématiques liées aux traces et à leur exploitation



Méthodologie Musette

- Définir le type de facilitation, définir le système observé
- Définir le modèle d'utilisation : principes
 - participation/validation de l'utilisateur
 - que chaque observable soit significatif pour l'utilisateur
 - qu'il puisse comprendre la trace primitive
 - que chaque observable soit inclus dans une signature tâche prévue a priori
 - associée à une utilisation pour l'utilisateur
 - que chaque observable soit effectivement observable
 - traitements raisonnables
- Définir en même temps
 - MU
 - Exemples de traces
 - Principes d'observation
 - EXTASI validantes



Modèle d'utilisation

- Entités
 - Constituants du système observé
 - logiciels utilisés
 - Certains éléments du modèle de conception des outils
 - Documents, requêtes, mot-clés, etc.
 - Entités qui n'existent que dans le cours de l'interaction avec le système
 - Focus utilisateur, groupe de documents,
- Événements
 - utilisateur se servant d'une fonctionnalité directement
 - ouvrir d'un nouveau document
 - utilisateur dans le cours de l'interaction
 - classer les résultats d'une requête
 - système lui-même
 - sauvegarde automatique



Questions d'éthique

- Pour chaque application
 - que l'utilisateur sache que des traces sont enregistrées
 - qu'il en maîtrise l'utilisation
 - possibilité de les voir
 - possibilité d'enlever des observables ou des parties (ex. attributs)
- Moyen simple
 - voir la trace se construire



Modèles utilisateurs et modèles d'utilisation

- Beaucoup de champs de recherche définissent des modèles utilisateurs
 - ensemble des propriétés associées à un utilisateur
 - utilisées pour personnaliser
 - Recherche
 - Documents (cf. Documents Virtuel Personnalisables)
- Limites des modèles utilisateurs
 - infiniment affinés pour répondre au besoins
 - en arriver à des modèles psycho-sociologiques ?
 - les besoins varient pour un même utilisateur
- Approche Musette
 - les modèles d'utilisations sont au moins aussi pertinents que les modèles utilisateurs
 - se concentrent sur la tâche en cours (réelle)
 - Interaction utilisateur / système



Auto-confrontation à la trace

- Traces
 - possibilité de lire et relire son histoire récente
 - ce n'est pas nouveau
 - historiques
 - relire son histoire de navigation
 - plus généralement,
 - toute tâche laisse des inscriptions matérielles plus ou moins volontaires qui seront à leur tour utilisées pour coordonner l'action (poursuite de la tâche en cours, « synchronisation » d'une tâche avec les tâches passées)
- Thème de recherche en soi
 - comment présenter ?
 - comment piloter la présentation ?
 - quels outils pour gérer sa trace ?
 - raconter une histoire vs percevoir des configurations
 - etc.
- SBT temps-réel



Collecte des traces SBT Agents en ligne

- Génèrent des sources de traçage
 - tracent l'utilisation pendant l'interaction
 - récupèrent les événements systèmes
 - doivent récupérer *toute* l'information nécessaire
 - ce qui est en mémoire à un instant donné pourra disparaître rapidement
- Deux types
 - embedded » (comme les journalistes)
 - nécessite d'intervenir dans le code de l'application
 - géré par le programmeur de/des l'application
 - hors-projet
 - dans le projet (idéal)
 - extérieur
 - agent de traçage qui tourne en même temps que le système dont on étudie l'interaction avec l'utilisateur
 - nécessite d'accéder au système
 - applications
 - système d'exploitation
 - impossible sans API d'interrogation



Collecte des traces SBT Agents hors-ligne

- Génèrent la trace première
 - plus ou moins complexes
 - traduction, réorganisation des sources de collecte
 - automatique
 - du temps pour les calculs
 - ... sauf si on est en temps-réel
 - manuel
 - Ex. annotation de vidéo, recueil de carnet d'observation
 - plus ou moins paramétrables
 - possibilité d'un agent générique ?
 - paramétrage par un fichier de configuration qui spécifie ce qui est intéressant ou non, les réécritures et transformation nécessaires
 - on n'en est pas encore là
 - agents *ad hoc*



Musette et l'analyse

- Trace d'utilisation
 - assistance à l'utilisateur fondée sur les traces
 - mais aussi outil d'analyse pour le chercheur
 - pourquoi ?
 - prise en compte du niveau de l'observateur
 - autonomisation de la trace
 - remarque
 - dans la lignée de l'accent mis sur les études d'usage au niveau national (laboratoires et plateformes d'études des usages / appels à projets)
- Musette-analyse
 - approche Musette pour l'analyse de traces, donc de tâches et d'activités
 - analyse seule
 - analyse en vue de / avec assistance



Musette-analyse : outils d'analyse de traces

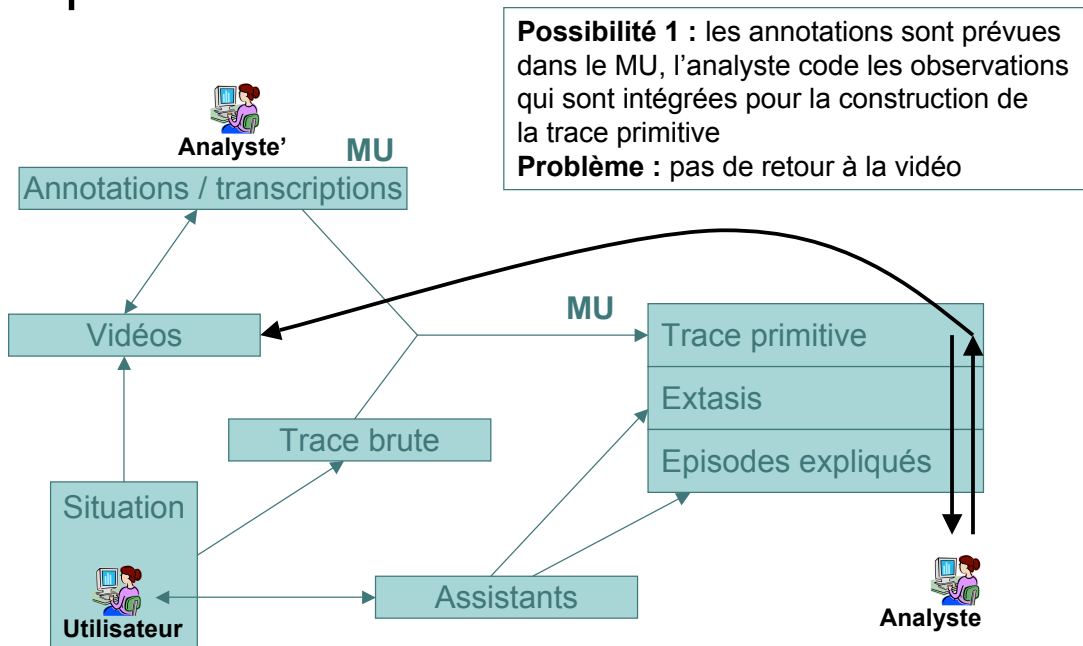
- Analyse directe de la trace
 - présentation / navigation
- La signature de tâche comme requête de l'analyste
 - vérifier les hypothèses initiales ayant présidé à la construction de la trace
 - construire et valider des hypothèses nouvelles
 - se rapproche des travaux orientés analyse assistée par ordinateur
 - linguistique de corpus particulièrement
- Autres possibilités
 - fouille de traces
 - recherche de motifs fréquents / non fréquents
 - avantage par rapport à fouille de données classique : la trace est déjà signifiante en soi
 - remettre en cause le passage de la trace brute à la trace primitive



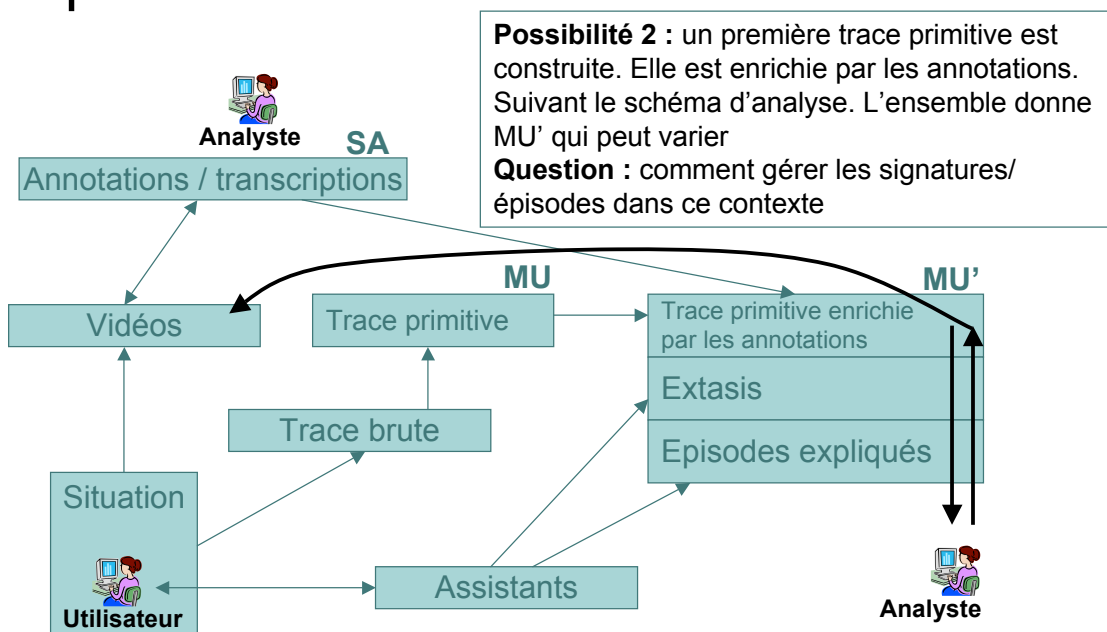
Musette-analyse : et la vidéo dans tout ça ?

- Un type d'observation que peut faire un analyste
 - utilisation d'un système, seul, en binôme
- Résulte déjà de choix, en fonction d'une problématique
 - choix de la situation
 - acteurs, moment, ...
 - placement de la (des) caméra(s)
 - cadrage (qui laisse le reste hors-champs)
 - analyse
 - transcription, annotation, construction d'une problématique
 - des similitudes avec Musette
 - mais une trace reste formelle
 - beaucoup moins riche que le magma d'une vidéo
 - moins riche
- La question
 - comment articuler l'utilisation de vidéo pour l'analyse avec une approche Musette
 - en tenant compte des particularités irréductibles de la vidéo

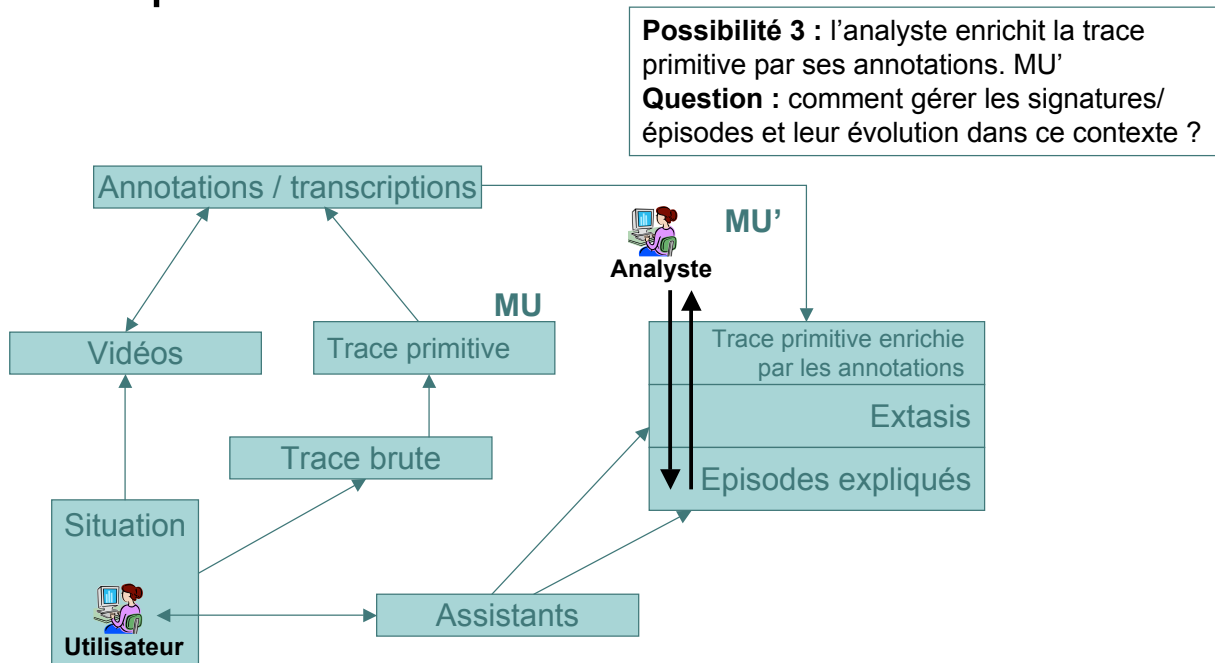
Musette-analyse : vers une vue globale



Musette-analyse : vers une vue globale



Musette-analyse : vers une vue globale



Musette-analyse : conclusion partielle

- Il reste du travail
- Vidéo et audio même combat
- A suivre...



Traces, expériences, connaissances

- L'utilisateur mobilise des connaissances dans le cadre de la réalisation de sa tâche
- Ces connaissances sont « inscrites » dans les supports qu'il manipule entre autres via un système informatique
 - elles sont ainsi inscrites en contexte
 - au moins en tant qu'indices
- Les traces de manipulation des supports d'inscription de connaissances « signent » leur contexte d'usage
- Les traces d'utilisation sont des conteneurs d'expériences de mise en évidence de connaissances



Assister la manipulation de la connaissance / expérience ?

- Dans le cas de tâches faiblement ou pas modélisées a priori
- En évoluant dynamiquement selon l'expérience d'utilisation captée
- En facilitant la réutilisation de l'expérience pour manipuler efficacement les inscriptions de la connaissance
- En facilitant le partage et la formation mutuelle sur la base des épisodes de manipulation de connaissance ainsi « concrétisée ».
- Emergence du sens dans la trace